```
EEEEEEEEEEEEEEE  DDDDDDDDDDD     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFFFFFFFFFFFFFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEEEEEEEEEEE     DDD         DDD  FFFFFFFFFFFF
EEEEEEEEEEEE     DDD         DDD  FFFFFFFFFFFF
EEEEEEEEEEEE     DDD         DDD  FFFFFFFFFFFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEE              DDD         DDD  FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDD     FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFF
```

```
Va
--
00
00
00
00
00
00
00
00
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
```

```
EEEEEEEEEE  DDDDDDDD    FFFFFFFFFF    AAAAAA      SSSSSSSS  KK      KK
EEEEEEEEEE  DDDDDDDD    FFFFFFFFFF    AAAAAA      SSSSSSSS  KK      KK
EE          DD      DD  FF          AA      AA  SS          KK      KK
EE          DD      DD  FF          AA      AA  SS          KK      KK
EE          DD      DD  FF          AA      AA  SS          KK    KK
EE          DD      DD  FF          AA      AA  SS          KK    KK
EEEEEEEE    DD      DD  FFFFFFFF    AA      AA    SSSSSS    KKKKKK
EEEEEEEE     D      DD  FFFFFFFF    AA      AA    SSSSSS    KKKKKK
EE          DD      DD  FF          AAAAAAAAAA          SS  KK    KK
EE          DD      DD  FF          AAAAAAAAAA          SS  KK    KK
EE          DD      DD  FF          AA      AA          SS  KK      KK      ....
EE          DD      DD  FF          AA      AA          SS  KK      KK      ....
EEEEEEEEEE  DDDDDDDD    FF          AA      AA  SSSSSSSS    KK      KK      ....
EEEEEEEEEE  DDDDDDDD    FF          AA      AA  SSSSSSSS    KK      KK      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0001            [ IDENT ('V04-000'),
0002        { ++
0003        ******************************************************************************
0004        **                                                                          *
0005        **  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0006        **  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0007        **  ALL RIGHTS RESERVED.                                                    *
0008        **                                                                          *
0009        **  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0010        **  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0011        **  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0012        **  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0013        **  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0014        **  TRANSFERRED.                                                            *
0015        **                                                                          *
0016        **  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0017        **  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0018        **  CORPORATION.                                                            *
0019        **                                                                          *
0020        **  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0021        **  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0022        **                                                                          *
0023        **                                                                          *
0024        ******************************************************************************
0025
0026
0027
0028
0029        FACILITY:       VAX/VMS EDF (EDIT/FDL) UTILITY
0030
0031        ABSTRACT:       This facility is used to create, modify, and optimize
0032                        FDL specification files.
0033
0034        ENVIRONMENT:    NATIVE/USER MODE
0035
0036        AUTHOR:         Ken F. Henderson Jr.
0037
0038        CREATION DATE:  27-Mar-1981
0039
0040        MODIFIED BY:
0041                        V03-018 JWT0191         Jim Teague              2 Aug 1984
0042                                Remove knowledge of ERASE_ON_DELETE.
0043
0044                        V03-017 RRB0017         Rowland R. Bradley      6 Mar 1984
0045                                Disallow ACLs - Commented out ACLs, will support
0046                                later.
0047
0048                        V03-016 RRB0009         Rowland R. Bradley     22 Jan 1984
0049                                Enhancement for display of # buckets in index, # of
0050                                pages to cache index, and average # key examinations.
0051
0052                        V03-015 RRB0007         Rowland R. Bradley     19 Jan 1984
0053                                Fix set analysis file to update the correct data
0054                                structure.
0055
0056                        V03-014 KFH0014         Ken Henderson          10 Sep 1983
0057                                Support for named UICs
```

```
0058
0059        V03-013 KFH0013         Ken Henderson            8 Aug 1983
0060                Bugfixing for FT1.
0061                Changes for seperate compilation.
0062
0063        V03-012 KFH0012         Ken Henderson           27 Jul 1983
0064                Fixed calculation of record and bucket
0065                overheads in blocks_in_bucket question.
0066                Added DEFERRED_WRITE.
0067
0068        V03-011 KFH0011         Ken Henderson           27 May 1983
0069                Modified PRE_PROCESS for KEY_COMP_WANTED,
0070                REC_COMP_WANTED, IDX_COMP_WANTED to
0071                force not wanted if not String datatype.
0072                Also force REC_COMP_WANTED to false if
0073                not Key 0.
0074
0075        V03-010 KFH0010         Ken Henderson           26 Apr 1983
0076                Modified PRE_PROCESS for
0077                EDF$K_NUMBER_KEYS, EDF$K_SURFACE_OPTION.
0078                Add ASK_KEY_SIZE, ASK_KEY_POSITION.
0079                Add ADD_KEY, DELETE_KEY to SCRIPT_OPTION.
0080                Removed EDF$K_GLOBAL_COUNT question.
0081
0082        V03-009 KFH0009         Ken Henderson           14 Apr 1983
0083                Changed max bucketsize to 63 from 65.
0084                Added SET_FUNCTION, GRANULARITY, PROMPTING,
0085                JOURNAL_ENABLED, and RESPONSES. Modified
0086                questions about DUPLICATES, COMPRESSION_WANTED.
0087
0088        V03-008 KFH0008         Ken Henderson            7 Mar 1983
0089                Changed max bucketsize to 65 from 127.
0090
0091        V03-007 KFH0007         Ken Henderson           20 Jan 1983
0092                Fixed REGIS support in DESIGN_CYCLE
0093                section of PRE_PROCESS. Also removed
0094                references to DASH. Also added
0095                Depthpoint displays to bucketsize
0096                question.
0097
0098        V03-006 KFH0006         Ken Henderson           22 Nov 1982
0099                Added support for additional FILE and
0100                CONNECT attributes.
0101
0102        V03-005 KFH0005         Ken Henderson           8 Sept 1982
0103                Modifed almost all variables to fit into
0104                new database scheme of arrays. Also added
0105                QUERY routine to process the QTAB table-
0106                driven Q+As. Also added support routines
0107                for QUERY. Also replaced almost ALL the
0108                "ASK_xxx" routines with QTAB/QUERY.
0109
0110        V03-004 KFH0004         Ken Henderson           19 April 1982
0111                Modified ASK_BUCKET to correct its
0112                handling of alternate keys.
0113
0114        V03-003 KFH0003         Ken Henderson           24-Mar-1982
```

```
0115                            Modified ASK_TEST_SECONDARY_VALUE to fix
0116                            QAR 833
0117
0118                    V03-002 KFH0002         Ken Henderson           23-Mar-1982
0119                            Modified several 'ASK_' routines to fix
0120                            FT2 QARs 745,746
0121
0122                    V03-001 KFH0001         Ken Henderson           17-Mar-1982
0123                            Modified several 'ASK_' routines to fix
0124                            FT2 QARs 509,449,574,575
0125
0126            -- }
```

```
0128            ENVIRONMENT ('LIB$:EDFASK'),
0129
0130            INHERIT (
0131
0132            'SYS$LIBRARY:STARLET',
0133            'SHRLIB$:FDLPARDEF',
0134            'LIB$:EDFSDLMSG',
0135            'LIB$:EDFSTRUCT',
0136            'LIB$:EDFCONST',
0137            'LIB$:EDFTYPE',
0138            'LIB$:EDFVAR',
0139            'LIB$:EDFEXTERN',
0140            'LIB$:EDFCHF'
0141            'LIB$:EDFUTIL'
0142
0143            )]
0144
0145            MODULE EDFASK (INPUT,OUTPUT);
```

```
0147            { ++
0148
0149            WRITE_HELP -- Routine to output per question help text.
0150
0151            This procedure cases on qtab-offset to output the help text.
0152
0153            CALLING SEQUENCE:
0154
0155            WRITE_HELP;
0156
0157            INPUT PARAMETERS:
0158
0159            none
0160
0161            IMPLICIT INPUTS:
0162
0163            none
0164
0165            OUTPUT PARAMETERS:
0166
0167            none
0168
0169            IMPLICIT OUTPUTS:
0170
0171            none
0172
0173            ROUTINES CALLED:
0174
0175            none
0176
0177            ROUTINE VALUE:
0178
0179            none
0180
0181            SIGNALS:
0182
0183            none
0184
0185            SIDE EFFECTS:
0186
0187
0188            -- }
```

```
0190        PROCEDURE WRITE_HELP;
0191
0192        BEGIN
0193
0194            CASE QTAB_OFFSET OF
0195
0196                EDF$K_DESIGN_CYCLE :
0197
0198                    WRITELN (SHIFT,'Type the 2 letter mnemonic of the selected option.');
0199
0200                EDF$K_KEY_POSITION :
0201
0202                    WRITELN (SHIFT,
0203                    'This is the starting byte of the key or key segment.');
0204
0205                EDF$K_KEY_DIST :
0206
0207                    WRITELN (
0208                    SHIFT,'This refers to records that are added to the',CRLF_SHIFT,
0209                    'file after it is initially loaded.');
0210
0211                EDF$K_KEY_CHANGES :
0212
0213                    WRITELN (SHIFT,'This enables or disables the RMS option.');
0214
0215                EDF$K_KEY_DUPS :
0216
0217                    WRITELN (SHIFT,'This enables or disables the RMS option.');
0218
0219                EDF$K_SEGMENTED :
0220
0221                    WRITELN (SHIFT,'Each string key may consist of up to 8 parts.');
0222
0223                EDF$K_GLOBAL_WANTED :
0224
0225                    WRITELN (
0226                    SHIFT,'These usually increase the speed of file sharing,',
0227                    CRLF_SHIFT,'at the expense of using more physical memory.');
0228
0229                EDF$K_RESPONSES :
0230
0231                    WRITELN (
0232                    SHIFT,'Automatic means the default answers will be used without',
0233                    CRLF_SHIFT,'waiting for confirmation.');
0234
0235                EDF$K_PROMPTING :
0236
0237                    WRITELN (SHIFT,'This controls whether full menus are displayed.');
0238
0239                EDF$K_NUMBER_RECORDS :
0240
0241                    WRITELN (SHIFT,'This will determine the allocation of the file.');
0242
0243                EDF$K_ASCENDING_LOAD :
0244
0245                    WRITELN (
0246                    SHIFT,'This refers to the order of the initial records loaded.');
```

```
0247
0248                EDF$K_INITIAL_COUNT,
0249                EDF$K_INITIAL_COUNT_LOW,
0250                EDF$K_INITIAL_COUNT_HIGH :
0251
0252                    WRITELN (SHIFT,
0253                    'These are the records initially loaded into the file.',CRLF_SHIFT,
0254                    'If the file will have no "Load" operation, specify "0".');
0255
0256                EDF$K_ADDED_COUNT,
0257                EDF$K_ADDED_COUNT_LOW,
0258                EDF$K_ADDED_COUNT_HIGH :
0259
0260                    WRITELN (SHIFT,
0261                    'These are the records added after the initial file load.');
0262
0263                EDF$K_BLOCK_SPAN :
0264
0265                    WRITELN (
0266                    SHIFT,'If no, each record plus overhead must fit in a disk block.',
0267                    CRLF_SHIFT,'Also, some space may be wasted at the end of blocks.');
0268
0269                EDF$K_KEY_LOW,
0270                EDF$K_KEY_HIGH,
0271                EDF$K_KEY_SIZE :
0272
0273                    WRITELN (SHIFT,'This is the length of the key (segment) in bytes.',
0274                    CRLF_SHIFT,
0275                    '(With multi-segment keys, answer "0" after the last segment.)');
0276
0277                EDF$K_PROLOGUE_VERSION :
0278
0279                    WRITELN (SHIFT,
0280                    'This refers to the structure level of the data file.',
0281                    CRLF_SHIFT,
0282                    'A value of 0 lets RMS choose an appropriate prolog.');
0283
0284                EDF$K_KEY_COMP_WANTED,
0285                EDF$K_REC_COMP_WANTED,
0286                EDF$K_IDX_COMP_WANTED :
0287
0288                    WRITELN (SHIFT,
0289                    'If an Analyze/RMS indicates little compression is acheived',
0290                    CRLF_SHIFT,
0291                    'then answer No, otherwise it is usually better to answer Yes.');
0292
0293                EDF$K_CLUSTER_SIZE :
0294
0295                    WRITELN (SHIFT,
0296                    'SHOW DEVICE/FULL can be used to determine this value.');
0297
0298                EDF$K_ASCENDING_ADDED :
0299
0300                    WRITELN (SHIFT,
0301                    'This refers to the orderring of additional records.');
0302
0303                EDF$K_BLOCKS_IN_BUCKET :
```

```
0304
0305                WRITELN (SHIFT,
0306                'Legal range is 1 to 63 blocks per bucket, and buckets must',
0307                CRLF_SHIFT,
0308                'be large enough to hold at least 1 record plus overhead.');
0309
0310            EDF$K_BUCKET_WEIGHT :
0311
0312                WRITELN (SHIFT,
0313                'Smaller_Buffers: less memory and RMS processing used',
0314                CRLF_SHIFT,
0315                'Flatter_Files:   fewer actual disk accesses needed');
0316
0317            EDF$K_LOAD_METHOD :
0318
0319                IF WAIT_HELP THEN
0320
0321                    WRITELN (SHIFT,
0322                    'Legal values: Fast_Convert, NoFast_Convert, RMS_Puts')
0323
0324                ELSE
0325
0326                    WRITELN (SHIFT,
0327                    'Fast_Convert:   using the VAX-11 Convert/Fast_Load option',
0328                    CRLF_SHIFT,
0329                    'NoFast_Convert: using the VAX-11 Convert/NoFast_Load option',
0330                    CRLF_SHIFT,
0331                    'RMS_Puts:       writing to a file from a High Level Language');
0332
0333            EDF$K_FILL_LOW,
0334            EDF$K_FILL_HIGH,
0335            EDF$K_DESIRED_FILL :
0336
0337                WRITELN (SHIFT,'This is the initial file loading fill factor.');
0338
0339            EDF$K_KEY_TYPE :
0340
0341            BEGIN
0342
0343                IF WAIT_HELP THEN
0344
0345                    WRITELN (SHIFT,
0346                    'Legal types: Bin2 Bin4 Bin8 Int2 Int4 Int8 Decimal String')
0347
0348                ELSE
0349
0350                    WRITELN (SHIFT,
0351                    'Use',
0352                    CRLF_SHIFT,
0353                    '"Binx" types for unsigned binary keys of 2, 4 or 8 bytes,',
0354                    CRLF_SHIFT,
0355                    '"Intx" types for signed binary key of 2, 4 or 8 bytes,',
0356                    CRLF_SHIFT,
0357                    '"Decimal" type for packed decimal key of 1 to 16 bytes,',
0358                    CRLF_SHIFT,
0359                    '"String" type for character string key of 1 to 255 bytes.');
0360
```

```
0361                END;    { EDF$K_KEY_TYPE }
0362
0363                EDF$K_RECORD_FORMAT :
0364
0365                    WRITELN (
0366                    SHIFT,'Indexed files are only Fixed or Variable.',CRLF_SHIFT,
0367                    'Stream format (Seq only) is Stream, Stream_CR, or Stream_LF.');
0368
0369                EDF$K_ACTIVE_KEY :
0370
0371                    WRITELN (SHIFT,'Select an already defined key.');
0372
0373                EDF$K_NUMBER_KEYS :
0374
0375                    WRITELN (SHIFT,'An Indexed file can have from 1 to 255 keys.');
0376
0377                EDF$K_CONTROL_SIZE :
0378
0379                    WRITELN (SHIFT,'This refers to the Fixed portion of the record.');
0380
0381                EDF$K_SIZE_LOW,
0382                EDF$K_SIZE_HIGH,
0383                EDF$K_MEAN_RECORD_SIZE :
0384
0385                    WRITELN (SHIFT,'This refers to the records in the data file.');
0386
0387                EDF$K_MAX_RECORD_SIZE :
0388
0389                BEGIN
0390
0391                    WRITELN (SHIFT,
0392                    'This sets the longest record that can be stored in the file.');
0393
0394                    IF IDATA[EDF$K_SCRIPT_OPTION] <> EDF$K_REL_DESIGN_FDL THEN
0395
0396                        WRITELN (SHIFT,
0397                        'A maximum of 0 will set no explicit maximum.');
0398
0399                END;    { EDF$K_MAX_RECORD_SIZE }
0400
0401                EDF$K_CARR_CTRL :
0402
0403                    WRITELN (SHIFT,'This sets the Record attributes of the file.');
0404
0405            OTHERWISE
0406
0407                { NULL-STATEMENT } ;
0408
0409            END;        { CASE }
0410
0411            IF (
0412            (WAIT_HELP)
0413            AND
0414            (NOT AUTO_TUNE)
0415            ) THEN
0416
0417                LIB$WAIT (3.0);
```

```
0418
0419            END;    { WRITE_HELP }
```

```
0421            { ++
0422
0423            WRITE_QUESTION -- Routine to output the question text.
0424
0425            This procedure cases on the qtab-offset and outputs the correct string.
0426
0427            CALLING SEQUENCE:
0428
0429            WRITE_QUESTION;
0430
0431            INPUT PARAMETERS:
0432
0433            none
0434
0435            IMPLICIT INPUTS:
0436
0437            none
0438
0439            OUTPUT PARAMETERS:
0440
0441            none
0442
0443            IMPLICIT OUTPUTS:
0444
0445            none
0446
0447            ROUTINES CALLED:
0448
0449            none
0450
0451            ROUTINE VALUE:
0452
0453            none
0454
0455            SIGNALS:
0456
0457            none
0458
0459            SIDE EFFECTS:
0460
0461
0462            -- }
```

```
0464            PROCEDURE WRITE_QUESTION;
0465
0466            BEGIN
0467
0468                CASE QTAB_OFFSET OF
0469
0470                    EDF$K_INITIAL_COUNT_LOW :
0471
0472                        WRITE (SHIFT,
0473                        'Low bound: Initial Load of Recs    (0-1Giga)[0]    : ');
0474
0475                     DF$K_INITIAL_COUNT_HIGH :
0476
0477                     GIN
0478
0479                        WRITE (SHIFT,'High bound: Initial Load of Recs(',
0480                        IDATA[EDF$K_Y_LOW]:NUM_LEN(IDATA[EDF$K_Y_LOW]),
0481                        '-1Giga)[',DEF:NUM_LEN(DEF),']');
0482
0483                        IF (NUM_LEN(IDATA[EDF$K_Y_LOW])+NUM_LEN(DEF)) <= 3 THEN
0484
0485                            WRITE ('         : ')
0486
0487                        ELSE
0488
0489                            WRITE (' : ');
0490
0491                    END;    { EDF$K_INITIAL_COUNT_HIGH }
0492
0493                    EDF$K_ADDED_COUNT_LOW :
0494
0495                    BEGIN
0496
0497                        WRITE (SHIFT,
0498                        'Low bound: Number of Added Recs    (0-1Giga)[0]    : ');
0499
0500                    END;    { EDF$K_ADDED_COUNT_LOW }
0501
0502                    EDF$K_ADDED_COUNT_HIGH :
0503
0504                    BEGIN
0505
0506                        WRITE (SHIFT,'High bound: Number of Added Recs(',
0507                        IDATA[EDF$K_Y_LOW]:NUM_LEN(IDATA[EDF$K_Y_LOW]),
0508                        '-1Giga)[',DEF:NUM_LEN(DEF),']');
0509
0510                        IF (NUM_LEN(IDATA[EDF$K_Y_LOW])+NUM_LEN(DEF)) <= 3 THEN
0511
0512                            WRITE ('         : ')
0513
0514                        ELSE
0515
0516                            WRITE (' : ');
0517
0518                    END;    { EDF$K_ADDED_COUNT_HIGH }
0519
0520                    EDF$K_KEY_LOW :
```

```
0521
0522                    WRITE (SHIFT,'Low bound: Key',
0523                    IDATA[EDF$K_ACTIVE_KEY]:3,
0524                    ' Length    (1-',
0525                    MAX_KEY_SIZE:NUM_LEN(MAX_KEY_SIZE),')[1]    : ');
0526
0527                EDF$K_KEY_HIGH :
0528
0529                    WRITE (SHIFT,'High bound: Key',
0530                    IDATA[EDF$K_ACTIVE_KEY]:3,' Length  (',
0531                    IDATA[EDF$K_Y_LOW]:NUM_LEN(IDATA[EDF$K_Y_LOW]),
0532                    '-',MAX_KEY_SIZE:NUM_LEN(MAX_KEY_SIZE),')[',
0533                    MAX_KEY_SIZE:NUM_LEN(MAX_KEY_SIZE),']       : ');
0534
0535                EDF$K_SIZE_LOW :
0536
0537                BEGIN
0538
0539                    WRITE (SHIFT,'Low bound: Record Size                (1-',
0540                    CUR_MAX_REC:5,')[1] : ');
0541
0542                END;    { EDF$K_SIZE_LOW }
0543
0544                EDF$K_SIZE_HIGH :
0545
0546                BEGIN
0547
0548                    WRITE (SHIFT,'High bound: Record Size               (',
0549                    IDATA[EDF$K_Y_LOW]:NUM_LEN(IDATA[EDF$K_Y_LOW]),'-',
0550                    CUR_MAX_REC:5,')[1000]');
0551
0552                    IF NUM_LEN(IDATA[EDF$K_Y_LOW]) < 3 THEN
0553
0554                        WRITE ('        : ')
0555
0556                    ELSE
0557
0558                        WRITE (' : ');
0559
0560                END;    { EDF$K_SIZE_HIGH }
0561
0562                EDF$K_FILL_LOW :
0563
0564                    WRITE (SHIFT,'Low bound: Key',
0565                    IDATA[EDF$K_ACTIVE_KEY]:3,
0566                    ' Init Fill %       (50-100)[50]    : ');
0567
0568                EDF$K_FILL_HIGH :
0569
0570                    WRITE (SHIFT,'High bound: Key',
0571                    IDATA[EDF$K_ACTIVE_KEY]:3,
0572                    ' Init Fill %       (50-100)[100]   : ');
0573
0574                EDF$K_SCRIPT_OPTION :
0575
0576                BEGIN
0577
```

```
0578                    CLEAR (IF_FULL_PROMPT);
0579
0580                    { +
0581                    Show the menu only if we're being verbose.
0582                    - }
0583                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
0584
0585                    BEGIN
0586
0587                        { +
0588                        Put the title out in reverse video.
0589                        - }
0590                        WRITELN (
0591                        SHIFT,
0592                        '                    ',
0593                        ANSI_REVERSE,
0594                        ' Script Title Selection ',
0595                        ANSI_RESET,
0596                        CRLF,
0597                        CRLF_SHIFT,
0598                'Add_Key            modeling and addition of a new index''s parameters',
0599                        CRLF_SHIFT,
0600                'Delete_Key removal of the highest index''s parameters',
0601                        CRLF_SHIFT,
0602                'Indexed            modeling of parameters for an entire Indexed file',
0603                        CRLF_SHIFT,
0604                'Optimize   tuning of all indices'' parameters using file statistics',
0605                        CRLF_SHIFT,
0606                'Relative   selection of parameters for a Relative file',
0607                        CRLF_SHIFT,
0608                'Sequential selection of parameters for a Sequential file',
0609                        CRLF_SHIFT,
0610                'Touchup            remodeling of parameters for a particular index',
0611                        CRLF
0612                        );
0613
0614                END
0615
0616                ELSE
0617
0618                    WRITELN (SHIFT,
0619                    '(Add_Key Delete_Key Indexed Optimize',
0620                    CRLF_SHIFT,
0621                    ' Relative Sequential Touchup)');
0622
0623                { +
0624                Pop the question.
0625                - }
0626                WRITE (SHIFT,'Editing Script Title          (Keyword)',
0627                ANSI_REVERSE,'[-]',ANSI_RESET,'     : ');
0628
0629            END;    { EDF$K_SCRIPT_OPTION }
0630
0631            EDF$K_RETURN :
0632
0633                WRITE (CRLF_SHIFT,ANSI_REVERSE,CONTINUE_TEXT,
0634                ANSI_RESET,'          ');
```

```
0635
0636            EDF$K_DESIGN_CYCLE :
0637
0638                WRITE (SHIFT,'(Type "FD" to Finish Design)',
0639                CRLF_SHIFT,
0640                'Which File Parameter       (Mnemonic)[refresh]     : ');
0641
0642            EDF$K_CURRENT_FUNCTION :
0643
0644            BEGIN
0645
0646                CLEAR (SCREEN);
0647
0648                { +
0649                Only show the menu if we're being verbose.
0650                - }
0651                IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
0652
0653                BEGIN
0654
0655                    { +
0656                    Show the header in reverse video, and then the
0657                    rest of the menu.
0658                    - }
0659                    WRITELN (
0660                    SHIFT,
0661                    '                ',
0662                    ANSI_REVERSE,
0663                    EDF_HEADER,
0664                    ANSI_RESET,
0665                    CRLF,
0666                    CRLF_SHIFT,
0667                'Add        to insert one or more lines into the FDL definition',
0668                    CRLF_SHIFT,
0669                'Delete     to remove one or more lines from the FDL definition',
0670                    CRLF_SHIFT,
0671                'Exit       to leave the FDL Editor after creating the FDL file',
0672                    CRLF_SHIFT,
0673                'Help       to obtain information about the FDL Editor',
0674                    CRLF_SHIFT,
0675                'Invoke     to initiate a script of related questions',
0676                    CRLF_SHIFT,
0677                'Modify     to change existing line(s) in the FDL definition',
0678                    CRLF_SHIFT,
0679                'Quit       to abort the FDL Editor with no FDL file creation',
0680                    CRLF_SHIFT,
0681                'Set        to specify FDL Editor characteristics',
0682                    CRLF_SHIFT,
0683                'View       to display the current FDL Definition',
0684                    CRLF);
0685
0686            END { IF TRUE FULL_PROMPT OR TEMP_FULL_PROMPT }
0687
0688            ELSE
0689
0690                WRITELN (SHIFT,
0691                '(Add Delete Exit Help Invoke Modify Quit Set View)');
```

```
0692
0693                    { +
0694                    Pop the question.
0695                    - }
0696                    WRITE (SHIFT,
0697                    'Main Editor Function                (Keyword)[Help] : ');
0698
0699                END;    { EDF$K_CURRENT_FUNCTION }
0700
0701
0702                EDF$K_RESPONSES :
0703
0704                BEGIN
0705
0706                    WRITE (SHIFT,'(Automatic Manual)',CRLF_SHIFT,
0707                    'Default responses in scripts        (Keyword)[Auto] : ');
0708
0709                END;
0710
0711                EDF$K_PROMPTING :
0712
0713                BEGIN
0714
0715                    WRITELN (SHIFT,'(Brief Full)');
0716                    WRITE (SHIFT,
0717                    'Prompting level for menus  (Keyword)[Full] : ');
0718
0719                END;
0720
0721                EDF$K_KEY_POSITION :
0722
0723                BEGIN
0724
0725                    WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
0726                    ' Position          ');
0727
0728                    IF BDATA[EDF$K_SEGMENTED] THEN
0729
0730                        WRITE ('SEG',SEGMENT_NUMBER:1);
0731
0732                    WRITE ('    (0-',
0733                    MAX_KEY_POSITION:NUM_LEN(MAX_KEY_POSITION),')[0]    : ');
0734
0735                END;    { EDF$K_KEY_POSITION }
0736
0737                EDF$K_KEY_DIST :
0738
0739                BEGIN
0740
0741                    WRITELN (SHIFT,
0742                    'Will Added Records be Distributed Evenly over the');
0743
0744                    IF NOT OPTIMIZING THEN
0745
0746                        WRITE (SHIFT,'Initial')
0747
0748                    ELSE
```

```
0749
0750                        WRITE (SHIFT,'Reloaded');
0751
0752                    WRITE (' Range of Pri Key Values');
0753
0754                    IF NOT OPTIMIZING THEN
0755
0756                        WRITE (TAB);
0757
0758                    WRITE ('(Yes/No)[No]        : ');
0759
0760                END;    { EDF$K_KEY_DIST }
0761
0762                EDF$K_KEY_CHANGES :
0763
0764                    WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
0765                    ' Changes allowed           (Yes/No)[Yes]   : ');
0766
0767                EDF$K_KEY_DUPS :
0768
0769                BEGIN
0770
0771                    WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
0772                    ' Duplicates allowed        (Yes/No)[');
0773
0774                    { +
0775                    The default for the primary key is NO, for alternates YES.
0776                    - }
0777                    IF IDATA[EDF$K_ACTIVE_KEY] = 0 THEN
0778
0779                        WRITE ('No]     : ')
0780
0781                    ELSE
0782
0783                        WRITE ('Yes]    : ');
0784
0785                END;    { EDF$K_KEY_DUPS }
0786
0787                EDF$K_SEGMENTED :
0788
0789                    WRITE (SHIFT,
0790                    'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
0791                    ' Segmentation desired      (Yes/No)[No]    : ');
0792
0793                EDF$K_GLOBAL_WANTED :
0794
0795                    WRITE (SHIFT,
0796                    'Global Buffers desired             (Yes/No)[No]    : ');
0797
0798                EDF$K_NUMBER_RECORDS :
0799
0800                    WRITE (SHIFT,'File Capacity in Records      (0-1Giga)',
0801                    ANSI_REVERSE,'[-]',ANSI_RESET,'     : ');
0802
0803                EDF$K_INITIAL_COUNT :
0804
0805                BEGIN
```

```
0806
0807            IF NOT OPTIMIZING THEN
0808
0809                WRITELN (SHIFT,
0810                'Number of Records that will be Initially Loaded')
0811
0812            ELSE
0813
0814                WRITELN (SHIFT,'Number of Records that will be Reloaded');
0815
0816            WRITE (SHIFT,'into the File','                        (0-1Giga)');
0817
0818            IF NOT OPTIMIZING THEN
0819
0820                WRITE (ANSI_REVERSE,'[-]',ANSI_RESET,'  : ')
0821
0822            ELSE
0823
0824            BEGIN
0825
0826                WRITE ('[',OLD_COUNT:NUM_LEN(OLD_COUNT),']');
0827
0828                IF NUM_LEN(OLD_COUNT) > 4 THEN
0829
0830                    WRITE (' : ')
0831
0832                ELSE
0833
0834                    WRITE ('    : ');
0835
0836            END;
0837
0838        END;    { EDF$K_INITIAL_COUNT }
0839
0840        EDF$K_LOAD_METHOD :
0841
0842        BEGIN
0843
0844            WRITELN (SHIFT,'(Fast_Convert NoFast_Convert RMS_Puts)');
0845
0846            IF NOT OPTIMIZING THEN
0847
0848                WRITE (SHIFT,'Initial File Load Method  ')
0849
0850            ELSE
0851
0852                WRITE (SHIFT,'File Reloading Method             ');
0853
0854            WRITE ('(Keyword)[Fast]     : ');
0855
0856        END;    { EDF$K_LOAD_METHOD }
0857
0858        EDF$K_ASCENDING_LOAD :
0859
0860        BEGIN
0861
0862            IF NOT OPTIMIZING THEN
```

```
0863
0864                WRITELN (SHIFT,
0865                'Will Initial Records Typically be Loaded in Order')
0866
0867            ELSE
0868
0869                WRITELN (SHIFT,
0870                'Will the Records be Reloaded Typically in Order');
0871
0872            WRITE (SHIFT,
0873            'by Ascending Primary Key   (Yes/No)[No]    : ');
0874
0875        END;    { EDF$K_ASCENDING_LOAD }
0876
0877        EDF$K_ADDED_COUNT :
0878
0879        BEGIN
0880
0881            WRITELN (SHIFT,
0882            'Number of Additional Records to be Added After');
0883
0884            IF NOT OPTIMIZING THEN
0885
0886                WRITE (SHIFT,'the Initial File Load')
0887
0888            ELSE
0889
0890                WRITE (SHIFT,'the Reloading the File');
0891
0892            WRITE ('            (0-1Giga)[0]    : ');
0893
0894        END;    { EDF$K_ADDED_COUNT }
0895
0896        EDF$K_KEY_COMP_WANTED :
0897
0898            WRITE (SHIFT,
0899            'Data Key Compression desired       (Yes/No)[Yes]   : ');
0900
0901        EDF$K_REC_COMP_WANTED :
0902
0903            WRITE (SHIFT,
0904            'Data Record Compression desired    (Yes/No)[Yes]   : ');
0905
0906        EDF$K_IDX_COMP_WANTED :
0907
0908            WRITE (SHIFT,
0909            'Index Compression desired  (Yes/No)[Yes]   : ');
0910
0911        EDF$K_CLUSTER_SIZE :
0912
0913            WRITE (SHIFT,
0914            'Target disk volume Cluster Size    (1-1Giga)[3]    : ');
0915
0916        EDF$K_BLOCK_SPAN :
0917
0918            WRITE (SHIFT,
0919            'Records can span disk blocks       (Yes/No)[Yes]   : ');
```

```
0920
0921            EDF$K_ASCENDING_ADDED :
0922
0923                WRITE (SHIFT,
0924                'Will Additional Records Typically be Added in',CRLF_SHIFT,
0925                'Order by Ascending Primary Key     (Yes/No)[No]    : ');
0926
0927            EDF$K_PROLOGUE_VERSION :
0928
0929                WRITE (SHIFT,
0930                'File Prolog Version                (0-3)[3]        : ');
0931
0932            EDF$K_KEY_SIZE :
0933
0934            BEGIN
0935
0936                WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
0937                ' Length             ');
0938
0939                IF BDATA[EDF$K_SEGMENTED] THEN
0940
0941                    WRITE ('SEG',SEGMENT_NUMBER:1);
0942
0943                WRITE ('    (',MIN_KEY_SIZE:NUM_LEN(MIN_KEY_SIZE),'-',
0944                MAX_KEY_SIZE:NUM_LEN(MAX_KEY_SIZE),')'
0945                ANSI_REVERSE,'[-]',ANSI_RESET,'     : ');
0946
0947            END;    { EDF$K_KEY_SIZE }
0948
0949            EDF$K_BLOCKS_IN_BUCKET :
0950
0951            BEGIN
0952
0953                WRITE (SHIFT,'Emphasis Used In Defining Default:    ( ');
0954
0955                IF IDATA[EDF$K_BUCKET_WEIGHT] = EDF$K_SMALLER_BUFFERS THEN
0956
0957                    WRITELN ('   Smaller_buffers )')
0958
0959                ELSE
0960
0961                    WRITELN ('   Flatter_files   )');
0962
0963                WRITELN (SHIFT,'Suggested Bucket Sizes:                    ( ',
0964                BREAKPOINT_LEFT:6,
0965                BREAKPOINT_MID:6,BREAKPOINT_RIGHT:6,' )');
0966
0967                WRITELN (SHIFT,'Number of Levels in Index:          ( ',
0968                DEPTHPOINT_LEFT:6,
0969                DEPTHPOINT_MID:6,DEPTHPOINT_RIGHT:6,' )');
0970
0971                WRITELN (SHIFT,'Number of Buckets in Index:         ( ',
0972                NUMPOINT_LEFT:6,
0973                NUMPOINT_MID:6,NUMPOINT_RIGHT:6,' )');
0974
0975                WRITELN (SHIFT,'Pages Required to Cache Index:              ( ',
0976                PAGEPOINT_LEFT:6,
```

```
0977                PAGEPOINT_MID:6,PAGEPOINT_RIGHT:6,' )');
0978
0979                WRITELN (SHIFT,'Processing Used to Search Index:    ( ',
0980                EXAMPOINT_LEFT:6,
0981                EXAMPOINT_MID:6,EXAMPOINT_RIGHT:6,' )');
0982
0983                WRITE (CRLF_SHIFT,'Key',
0984                IDATA[EDF$K_ACTIVE_KEY]:3,
0985                ' Bucket Size                ',MIN_BUCKET:NUM_LEN(MIN_BUCKET),
0986                '-63)[',
0987                QTAB[QTAB_OFFSET].DEFAULT:NUM_LEN(QTAB[QTAB_OFFSET].DEFAULT),
0988                '] : ');
0989
0990            END;    { EDF$K_BLOCKS_IN_BUCKET }
0991
0992            EDF$K_BUCKET_WEIGHT :
0993
0994            BEGIN
0995
0996                WRITE (SHIFT,'(Smaller_Buffers Flatter_Files)',
0997                CRLF_SHIFT,'Emphasis for Default Bucket_Size(Keyword)[');
0998
0999                IF QTAB[QTAB_OFFSET].DEFAULT = EDF$K_FLATTER_FILES THEN
1000
1001                    WRITE ('Flat]   : ')
1002
1003                ELSE
1004
1005                    WRITE ('Small] : ');
1006
1007            END;    { EDF$K_BUCKET_WEIGHT }
1008
1009            EDF$K_DESIRED_FILL :
1010
1011                WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
1012                ' Load Fill Percent (50-100)[100]   :');
1013
1014            EDF$K_CONFIRM :
1015
1016                WRITE (SHIFT,
1017                'Replace this existing secondary    (Yes/No)[No]    : ');
1018
1019            EDF$K_DATA_FILE_NAME :
1020
1021                WRITE (SHIFT,
1022                'Data File file-spec                (1-126 chars)[null]',
1023                CRLF_SHIFT,': ');
1024
1025            EDF$K_ANALYSIS :
1026
1027                WRITE (SHIFT,'Analysis File file-spec       (1-126 chars)[null]',
1028                CRLF_SHIFT,': ');
1029
1030            EDF$K_OUTPUT :
1031
1032                WRITE (SHIFT,
1033                'Output File file-spec              (1-126 chars)[null]',
```

```
1034                        CRLF_SHIFT,': ');
1035
1036                    EDF$K_FDL_TITLE :
1037
1038                        WRITE (SHIFT,
1039                        'Text for FDL Title Section (1-126 chars)[null]',
1040                        CRLF_SHIFT,': ');
1041
1042                    EDF$K_KEY_NAME :
1043
1044                        WRITE (SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
1045                        ' Name                          (1-32 chars)[null]',CRLF_SHIFT,
1046                        ': ');
1047
1048                    EDF$K_KEY_TYPE :
1049
1050                    BEGIN
1051
1052                        WRITE (SHIFT,'(Bin2 Bin4 Bin8 Int2 Int4 Int8 Decimal String)',
1053                        CRLF_SHIFT,'Key',IDATA[EDF$K_ACTIVE_KEY]:3,
1054                        ' Data Type           (Keyword)[Str]  : ');
1055
1056                    END;    { EDF$K_KEY_TYPE }
1057
1058                    EDF$K_ACTIVE_KEY :
1059
1060                        WRITE (SHIFT,'Key of Reference                (',
1061                        LOW_KEY:NUM_LEN(LOW_KEY),'-',HIGH_KEY:NUM_LEN(HIGH_KEY)
1062                        ,')[0]      : ');
1063
1064                    EDF$K_NUMBER_KEYS :
1065
1066                    BEGIN
1067
1068                        WRITE (SHIFT,'Number of Keys to Define      (1-255)[',
1069                        QTAB[QTAB_OFFSET].DEFAULT:NUM_LEN(QTAB[QTAB_OFFSET].DEFAULT),
1070                        ']  : ');
1071
1072                    END;
1073
1074                    EDF$K_CARR_CTRL :
1075
1076                    BEGIN
1077
1078                        WRITE (SHIFT,'(Carriage_Return FORTRAN None Print)',
1079                        CRLF_SHIFT,'Carriage Control                  (Keyword)[Carr] : ');
1080
1081                    END;    { EDF$K_CARR_CTRL }
1082
1083                    EDF$K_RECORD_FORMAT :
1084
1085                    BEGIN
1086
1087                        CASE IDATA[EDF$K_SCRIPT_OPTION] OF
1088
1089                            EDF$K_ADD_KEY_FDL,
1090                            EDF$K_DELETE_KEY_FDL,
```

```
1091                    EDF$K_REDESIGN_FDL,
1092                    EDF$K_OPTIMIZE_FDL,
1093                    EDF$K_IDX_DESIGN_FDL : WRITELN (SHIFT,'(Fixed ,ariable)');
1094                    EDF$K_REL_DESIGN_FDL : WRITELN (SHIFT,'(Fix  , Variable VFC)');
1095                    EDF$K_SEQ_DESIGN_FDL : WRITELN (SHIFT,
1096                            '(Fixed Stream _CR _LF Undefined Variable VFC)');
1097
1098                OTHERWISE
1099
1100                    { NULL-STATEMENT } ;
1101
1102                END;        { CASE }
1103
1104                WRITE (SHIFT,
1105                'Record Format                          (Keyword)[Var]  : ');
1106
1107            END;    { EDF$K_RECORD_FORMAT }
1108
1109            EDF$K_CONTROL_SIZE :
1110
1111                WRITE (SHIFT,'Control Field Size            (1-',
1112                CUR_MAX_FIXED:NUM_LEN(CUR_MAX_FIXED),')[2]  : ');
1113
1114            EDF$K_MEAN_RECORD_SIZE :
1115
1116            BEGIN
1117
1118                WRITE (SHIFT);
1119
1120                IF VARIABLE_RECORDS THEN
1121
1122                    WRITE ('Mean ');
1123
1124                WRITE ('Record Size');
1125
1126                IF IDATA[EDF$K_RECORD_FORMAT] = FDL$C_VFC THEN
1127
1128                    WRITE (' w/fix');
1129
1130                IF NOT VARIABLE_RECORDS THEN
1131
1132                    WRITE (TAB);
1133
1134                WRITE ('            (1-',
1135                CUR_MAX_REC:NUM_LEN(CUR_MAX_REC),')',
1136                ANSI_REVERSE,'[=]',ANSI_RESET,'     : ');
1137
1138            END;    { EDF$K_MEAN_RECORD_SIZE }
1139
1140            EDF$K_SURFACE_OPTION :
1141
1142            BEGIN
1143
1144                { +
1145                See which surface.
1146                - }
1147                CLEAR (IF_FULL_PROMPT);
```

```
1148
1149                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
1150
1151                    BEGIN
1152
1153                        WRITELN (
1154                        SHIFT,
1155                        '                ',
1156                        ANSI_REVERSE,
1157                        ' Key',IDATA[EDF$K_ACTIVE_KEY]:3,' Graph Type Selection ',
1158                        ANSI_RESET,
1159                        CRLF,
1160                        CRLF_SHIFT,
1161            'Line       Bucket Size vs Index Depth      as a 2 dimensional plot',
1162                        CRLF_SHIFT,
1163            'Fill       Bucket Size vs     Load Fill Percent     vs Index Depth',
1164                        CRLF_SHIFT,
1165            'Key        Bucket Size vs         Key Length        vs Index Depth'
1166                        );
1167
1168                        IF IDATA[EDF$K_ACTIVE_KEY] = 0 THEN
1169
1170                        BEGIN
1171
1172            WRITELN (SHIFT,
1173            'Record Bucket Size vs        Record Size        vs Index Depth',
1174            CRLF_SHIFT,
1175            'Init   Bucket Size vs Initial Load Record Count vs Index Depth',
1176            CRLF_SHIFT,
1177            'Add    Bucket Size vs  Additional Record Count  vs Index Depth');
1178
1179                        END;
1180
1181                        WRITELN;
1182
1183                    END
1184
1185                    ELSE
1186
1187                    BEGIN
1188
1189                        WRITE (SHIFT,'(Line Fill Key');
1190
1191                        IF IDATA[EDF$K_ACTIVE_KEY] = 0 THEN
1192
1193                            WRITE (' Record Init Add)')
1194
1195                        ELSE
1196
1197                            WRITE (')');
1198
1199                        WRITELN;
1200
1201                    END;        { IF FULL_PROMPT OR TEMP_FULL_PROMPT }
1202
1203                    { *
1204                    Always ask the question, even for brief prompting.
```

```
1205                    - }
1206                    WRITE (SHIFT,'Graph type to display         (Keyword)[');
1207
1208                    CASE QTAB[QTAB_OFFSET].DEFAULT OF
1209
1210                        EDF$K_LINE_SURFACE :    WRITE ('Line]   : ');
1211                        EDF$K_FILL_SURFACE :    WRITE ('Fill]   : ');
1212                        EDF$K_KEY_SURFACE :     WRITE ('Key]    : ');
1213                        EDF$K_SIZE_SURFACE :    WRITE ('Rec]    : ');
1214                        EDF$K_INIT_SURFACE :    WRITE ('Init]   : ');
1215                        EDF$K_ADDED_SURFACE :   WRITE ('Add]    : ');
1216
1217                    OTHERWISE
1218
1219                        { NULL-STATEMENT } ;
1220
1221                    END;        { CASE }
1222
1223                END;    { EDF$K_SURFACE_OPTION }
1224
1225                EDF$K_GRANULARITY :
1226
1227                BEGIN
1228
1229                    { +
1230                    See what level of granularity.
1231                    - }
1232                    CLEAR (IF_FULL_PROMPT);
1233
1234                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
1235
1236                    BEGIN
1237
1238                        WRITELN (
1239                        SHIFT,
1240                        '                ',
1241                        ANSI_REVERSE,
1242                        ' Area Granularity Selection ',
1243                        ANSI_RESET);
1244
1245                        IF DEC_CRT THEN
1246
1247                        BEGIN
1248
1249                        WRITELN (CRLF,LOW_SHIFT,
1250            '   '(27)')0'(14)'lqqqqqqqqqqqqqk'(15)'    '(14)'lqqqqqqqqqqqqqk'(15)'    '(14)'lqqqqqqqqqqqqqk'(15)'    '(14)'lq
1251                        CRLF,LOW_SHIFT,
1252            ' 0 '(14)'x'(15)' Key 0 Data  '(14)'x'(15)'   0 '(14)'x'(15)' Key 0 Data  '(14)'x'(15)'   0 '(14)'x'(15)' Key 0 Data
1253                        CRLF,LOW_SHIFT,
1254            '   '(14)'x'(15)'             '(14)'x'(15)'     '(14)'tqqqqqqqqqqqqqu'(15)'    '(14)'tqqqqqqqqqqqqqu'(15)'    '(14
1255                        CRLF,LOW_SHIFT,
1256            '   '(14)'x'(15)' Key 0 Index '(14)'x'(15)'   1 '(14)'x'(15)' Key 0 Index '(14)'x'(15)'   1 '(14)'x'(15)' Key 0 Inde
1257                        CRLF,LOW_SHIFT,
1258            '   '(14)'x'(15)'             '(14)'x'(15)'     '(14)'x'(15)'             '(14)'x'(15)'    '(14)'tqqqqqqqqqqqqqu'(1
1259                        CRLF,LOW_SHIFT,
1260            '   '(14)'x'(15)' Key n Data  '(14)'x'(15)'     '(14)'x'(15)' Key n Data  '(14)'x'(15)'   2 '(14)'x'(15)' Key n Data
1261                        CRLF,LOW_SHIFT,
```

```
1262            '   '(14)'x'(15)'              '(14)'x'(15)'     '(14)'x'(15)'              '(14)'x'(15)'     '(14)'x'(15)'
1263                    CRLF,LOW_SHIFT,
1264            '   '(14)'x'(15)' Key n Index '(14)'x'(15)'     '(14)'x'(15)' Key n Index '(14)'x'(15)'     '(14)'x'(15)' Key n Inde
1265                    CRLF,LOW_SHIFT,
1266            '   '(14)'mqqqqqqqqqqqqqj'(15)'     '(14)'mqqqqqqqqqqqqqj'(15)'     '(14)'mqqqqqqqqqqqqqj'(15)'     '(14)'mqqqqqqqqq
1267                    CRLF,LOW_SHIFT,
1268            '       One (1)                Two (2)            Three (3)            Four (4)',
1269                    CRLF)
1270
1271                    END     { IF DEC_CRT }
1272
1273                    ELSE
1274
1275                    BEGIN
1276
1277                    WRITELN (CRLF,LOW_SHIFT,
1278            '   +-------------+     +-------------+     +-------------+     +-------------+',
1279                    CRLF,LOW_SHIFT,
1280            ' 0 ! Key 0 Data  !   0 ! Key 0 Data  !   0 ! Key 0 Data  !   0 ! Key 0 Data  !',
1281                    CRLF,LOW_SHIFT,
1282            '   !             !     +-------------+     +-------------+     +-------------+',
1283                    CRLF,LOW_SHIFT,
1284            '   ! Key 0 Index !   1 ! Key 0 Index !   1 ! Key 0 Index !   1 ! Key 0 Index !',
1285                    CRLF,LOW_SHIFT,
1286            '   !             !     !             !     +-------------+     +-------------+',
1287                    CRLF,LOW_SHIFT,
1288            '   ! Key n Data  !     ! Key n Data  !   2 ! Key n Data  !   2 ! Key n Data  !',
1289                    CRLF,LOW_SHIFT,
1290            '   !             !     !             !     !             !     +-------------+',
1291                    CRLF,LOW_SHIFT,
1292            '   ! Key n Index !     ! Key n Index !     ! Key n Index !   3 ! Key n Index !',
1293                    CRLF,LOW_SHIFT,
1294            '   +-------------+     +-------------+     +-------------+     +-------------+',
1295                    CRLF,LOW_SHIFT,
1296            '       One (1)                Two (2)            Three (3)            Four (4)',
1297                    CRLF);
1298
1299                    END;    { IF NOT DEC_CRT }
1300
1301                END
1302
1303                ELSE
1304
1305                BEGIN
1306
1307                    WRITELN (SHIFT,'(One Two Three Four Double)');
1308
1309                END;        { IF FULL_PROMPT OR TEMP_FULL_PROMPT }
1310
1311                { +
1312                Always ask the question, even for brief prompting.
1313                - }
1314                WRITELN (SHIFT,'(Type "Double" to allocate 2 areas per key)');
1315                WRITE (SHIFT,
1316                'Number of areas to allocate        (keyword)[Three] : ');
1317
1318            END;    { EDF$K_GRANULARITY }
```

```
1319
1320            EDF$K_SET_FUNCTION :
1321
1322            BEGIN
1323
1324                {+
1325                See what char to set.
1326                -}
1327                CLEAR (IF_FULL_PROMPT);
1328
1329                IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
1330
1331                BEGIN
1332
1333                    WRITELN (
1334                    SHIFT,
1335                    '                ',
1336                    ANSI_REVERSE,
1337                    ' FDL Editor SET Function ',
1338                    ANSI_RESET,
1339                    CRLF,
1340                    CRLF_SHIFT,
1341                    'Analysis       filespec of FDL Analysis file',
1342                    CRLF_SHIFT,
1343                    'Display                type of graph to display',
1344                    CRLF_SHIFT,
1345                    'Emphasis       of default bucketsize calculations',
1346                    CRLF_SHIFT,
1347                    'Granularity    number of areas in Indexed files',
1348                    CRLF_SHIFT,
1349                    'Number_Keys    number of keys in Indexed files',
1350                    CRLF_SHIFT,
1351                    'Output         filespec of FDL Output file',
1352                    CRLF_SHIFT,
1353                    'Prompting      Full or Brief prompting of menus',
1354                    CRLF_SHIFT,
1355                    'Responses      usage of default responses in scripts',
1356                    CRLF);
1357
1358            END
1359
1360            ELSE
1361
1362            BEGIN
1363
1364                WRITELN (SHIFT,
1365                '(Analysis Display Emphasis Granularity',
1366                CRLF_SHIFT,' Number_Keys Output Prompting Responses)');
1367
1368            END;        { IF FULL_PROMPT OR TEMP_FULL_PROMPT }
1369
1370            {+
1371            Always ask the question, even for brief prompting.
1372            -}
1373            WRITE (SHIFT,
1374            'Editor characteristic to set      (keyword)',
1375            ANSI_REVERSE,'[-]',ANSI_RESET,'     : ');
```

```
1376
1377                END;    { EDF$K_SET_FUNCTION }
1378
1379                EDF$K_MAX_RECORD_SIZE :
1380
1381                BEGIN
1382
1383                    WRITE (SHIFT,'Maximum Record Size           (');
1384
1385                    IF IDATA[EDF$K_SCRIPT_OPTION] = EDF$K_REL_DESIGN_FDL THEN
1386
1387                    BEGIN
1388
1389                        EXTRA   := 0;
1390                        WRITE (
1391                        LOWMAX:NUM_LEN(LOWMAX),'-',
1392                        CUR_MAX_REC:NUM_LEN(CUR_MAX_REC),')',
1393                        ANSI_REVERSE,'[=]',ANSI_RESET);
1394                        QTAB[QTAB_OFFSET].DEFAULT_OK    := FALSE;
1395
1396                    END
1397
1398                    ELSE
1399
1400                    BEGIN
1401
1402                        EXTRA   := 2;
1403                        WRITE ('0,',
1404                        LOWMAX:NUM_LEN(LOWMAX),'-',CUR_MAX_REC:NUM_LEN(CUR_MAX_REC),
1405                        ')[0]');
1406                        QTAB[QTAB_OFFSET].DEFAULT_OK    := TRUE;
1407                        QTAB[QTAB_OFFSET].DEFAULT       := 0;
1408
1409                    END;
1410
1411                    IF (
1412                    ( EXTRA + NUM_LEN(LOWMAX) + NUM_LEN(CUR_MAX_REC) ) > 9
1413                    ) THEN
1414
1415                        WRITE (' : ')
1416
1417                    ELSE
1418
1419                        WRITE ('         : ');
1420
1421                END;    { EDF$K_MAX_RECORD_SIZE }
1422
1423                EDF$K_TEST_PRIMARY :
1424
1425                BEGIN
1426
1427                    CLEAR (IF_FULL_PROMPT);
1428
1429                    IF FULL_CHOICE THEN
1430
1431                    BEGIN
1432
```

```
1433                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
1434
1435                    BEGIN
1436
1437            WRITELN (SHIFT,'                ',
1438            ANSI_REVERSE,
1439            ' Legal Primary Attributes ',
1440            ANSI_RESET,
1441            CRLF,
1442            CRLF_SHIFT,
1443            'ACCESS attributes set the run-time access mode of the file',
1444            CRLF_SHIFT,
1445            'AREA x attributes define the characteristics of file area x',
1446            CRLF_SHIFT,
1447            'CONNECT        attributes set various RMS run-time options',
1448            CRLF_SHIFT,
1449            'DATE   attributes set the date parameters of the file',
1450            CRLF_SHIFT,
1451            'FILE   attributes affect the entire RMS data file',
1452            CRLF_SHIFT,
1453            'JOURNAL        attributes set the journaling parameters of the file',
1454            CRLF_SHIFT,
1455            'KEY y  attributes define the characteristics of key y',
1456            CRLF_SHIFT,
1457            'RECORD attributes set the non-key aspects of each record',
1458            CRLF_SHIFT,
1459            'SHARING        attributes set the run-time sharing mode of the file',
1460            CRLF_SHIFT,
1461            'SYSTEM attributes document operating system-specific items',
1462            CRLF_SHIFT,
1463            'TITLE  is the header line for the FDL file',
1464            CRLF);
1465
1466            (* insert in above to handle ACLs
1467            'ACL    entries specify the Access-Control-List of the file',
1468            CRLF_SHIFT,
1469            *)
1470
1471                        END
1472
1473                        ELSE
1474                            WRITELN (SHIFT,
1475                            '(ACCESS AREA CONNECT DATE FILE JOURNAL',
1476                            CRLF_SHIFT,
1477                            ' KEY RECORD SHARING SYSTEM TITLE)');
1478
1479            (* ACL insert in above to handle ACLs *)
1480
1481                    END
1482
1483                    ELSE
1484
1485                    BEGIN
1486
1487                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
1488
1489                        BEGIN
```

```
1490
1491                        WRITELN (
1492                        SHIFT,'                ',
1493                        ANSI_REVERSE,
1494                        ' Current Primary Attributes ',
1495                        ANSI_RESET,
1496                        CRLF
1497                        );
1498
1499                        { +
1500                        Setup to display definition on the terminal.
1501                        - }
1502                        OPEN        (FDL_DEST,SYS$OUTPUT_NAME,NEW,
1503                                    RECORD_LENGTH := 252);
1504                        REWRITE     (FDL_DEST);
1505
1506                        SHOW_ALL_PRIMARIES;
1507
1508                        CLOSE       (FDL_DEST);
1509
1510                    END
1511
1512                    ELSE
1513
1514                        WRITELN (SHIFT,
1515                        '(Type "?" for a list of existing Primary Attributes)');
1516
1517                END;
1518
1519                { +
1520                Pop the question.
1521                - }
1522                WRITE (SHIFT,'Enter Desired Primary          (Keyword)[',
1523                DEFAULT_PRIMARY:PRIMARY_WIDTH[DEFAULT_PRIMARY]);
1524
1525                IF DEFAULT_PRIMARY IN [ AREA, KEY ] THEN
1526
1527                    WRITE (' ',DEFAULT_PRINUM:NUM_LEN(DEFAULT_PRINUM));
1528
1529                WRITE ('] : ');
1530
1531            END;    { EDF$K_TEST_PRIMARY }
1532
1533        OTHERWISE
1534
1535            { NULL-STATEMENT } ;
1536
1537        END;        { CASE }
1538
1539    END;    { WRITE_QUESTION }
```

```
1541            { ++
1542
1543            SPREAD_LOW_HIGH -- Routine to make sure high-bound is 5 away from low-bound.
1544
1545            This procedure adjusts Y_LOW,Y_HIGH until high-bound >= low-bound.
1546
1547            CALLING SEQUENCE:
1548
1549            SPREAD_LOW_HIGH (LO_LIM,HI_LIM);
1550
1551            INPUT PARAMETERS:
1552
1553            LO_LIM
1554            HI_LIM
1555
1556            IMPLICIT INPUTS:
1557
1558            none
1559
1560            OUTPUT PARAMETERS:
1561
1562            none
1563
1564            IMPLICIT OUTPUTS:
1565
1566            none
1567
1568            ROUTINES CALLED:
1569
1570            none
1571
1572            ROUTINE VALUE:
1573
1574            none
1575
1576            SIGNALS:
1577
1578            none
1579
1580            SIDE EFFECTS:
1581
1582
1583            -- }
```

```
1585        PROCEDURE SPREAD_LOW_HIGH ( LO_LIM, HI_LIM : INTEGER );
1586
1587        BEGIN
1588
1589            { +
1590            Make sure the bounds are at least 5 apart.
1591            - }
1592            IF (IDATA[EDF$K_Y_HIGH] - IDATA[EDF$K_Y_LOW]) < 5 THEN
1593
1594            BEGIN
1595
1596                REPEAT
1597
1598                    IF IDATA[EDF$K_Y_LOW] > LO_LIM THEN
1599
1600                        IDATA[EDF$K_Y_LOW]      := IDATA[EDF$K_Y_LOW] - 1;
1601
1602                    IF IDATA[EDF$K_Y_HIGH] < HI_LIM THEN
1603
1604                        IDATA[EDF$K_Y_HIGH]     := IDATA[EDF$K_Y_HIGH] + 1;
1605
1606                UNTIL (IDATA[EDF$K_Y_HIGH] - IDATA[EDF$K_Y_LOW]) > 4;
1607
1608            END;
1609
1610        END;    { SPREAD_LOW_HIGH }
```

```
1612            { ++
1613
1614            AUTO_SCALE -- Scale a surface plot.
1615
1616            This procedure adjusts Y_LOW,Y_HIGH,Y_INCR until the plot fits on the screen.
1617
1618            CALLING SEQUENCE:
1619
1620            AUTO_SCALE (LOW_LIMIT,HIGH_LIMIT);
1621
1622            INPUT PARAMETERS:
1623
1624            LOW_LIMIT
1625            HIGH_LIMIT
1626
1627            IMPLICIT INPUTS:
1628
1629            none
1630
1631            OUTPUT PARAMETERS:
1632
1633            none
1634
1635            IMPLICIT OUTPUTS:
1636
1637            none
1638
1639            ROUTINES CALLED:
1640
1641            none
1642
1643            ROUTINE VALUE:
1644
1645            none
1646
1647            SIGNALS:
1648
1649            none
1650
1651            SIDE EFFECTS:
1652
1653
1654            -- }
```

```
1656            PROCEDURE AUTO_SCALE ( LOW_LIMIT, HIGH_LIMIT : INTEGER );
1657
1658            BEGIN
1659
1660                { +
1661                Figure out what the step between lines should be.
1662                We always have max_array_row steps.
1663                - }
1664                TEMP_INT2   := IDATA[EDF$K_Y_HIGH];
1665                TEMP_REAL   := (IDATA[EDF$K_Y_HIGH] - IDATA[EDF$K_Y_LOW]) / MAX_ARRAY_ROW;
1666
1667                IDATA[EDF$K_Y_INCR] := TRUNC (TEMP_REAL);
1668
1669                IF ((IDATA[EDF$K_Y_HIGH] - IDATA[EDF$K_Y_LOW]) MOD MAX_ARRAY_ROW) > 0 THEN
1670
1671                    IDATA[EDF$K_Y_INCR]     := IDATA[EDF$K_Y_INCR] + 1;
1672
1673                { +
1674                Juggle the margins and the step until we get it to fit.
1675                - }
1676                REPEAT
1677
1678                    IDATA[EDF$K_Y_HIGH]     := IDATA[EDF$K_Y_LOW]
1679                                               + (MAX_ARRAY_ROW * IDATA[EDF$K_Y_INCR]);
1680
1681                    { +
1682                    Adjust down if too high.
1683                    - }
1684                    WHILE IDATA[EDF$K_Y_HIGH] > HIGH_LIMIT DO
1685
1686                    BEGIN
1687
1688                        IDATA[EDF$K_Y_LOW]          := IDATA[EDF$K_Y_LOW] - 1;
1689                        IDATA[EDF$K_Y_HIGH]         := IDATA[EDF$K_Y_HIGH] - 1;
1690
1691                    END;
1692
1693                    { +
1694                    Adjust up if too low.
1695                    - }
1696                    WHILE IDATA[EDF$K_Y_LOW] < LOW_LIMIT DO
1697
1698                    BEGIN
1699
1700                        IDATA[EDF$K_Y_LOW]          := IDATA[EDF$K_Y_LOW] + 1;
1701                        IDATA[EDF$K_Y_HIGH]         := IDATA[EDF$K_Y_HIGH] + 1;
1702
1703                    END;
1704
1705                    { +
1706                    Try a smaller step if this didn't work.
1707                    - }
1708                    IF (IDATA[EDF$K_Y_LOW] < LOW_LIMIT) OR (IDATA[EDF$K_Y_HIGH] > HIGH_LIMIT) THEN
1709
1710                        IDATA[EDF$K_Y_INCR]         := IDATA[EDF$K_Y_INCR] - 1;
1711
1712                UNTIL (IDATA[EDF$K_Y_LOW] >= LOW_LIMIT) AND (IDATA[EDF$K_Y_HIGH] <= HIGH_LIMIT);
```

```
1713
1714                {+
1715                Adjust the range up if we've squeezed it low.
1716                -}
1717                IF IDATA[EDF$K_Y_HIGH] < TEMP_INT2 THEN
1718
1719                BEGIN
1720
1721                    IDATA[EDF$K_Y_LOW]      := IDATA[EDF$K_Y_LOW]
1722                                                + (TEMP_INT2 - IDATA[EDF$K_Y_HIGH]);
1723                    IDATA[EDF$K_Y_HIGH]     := IDATA[EDF$K_Y_HIGH]
1724                                                + (TEMP_INT2 - IDATA[EDF$K_Y_HIGH]);
1725
1726                END;
1727
1728            END;    { AUTO_SCALE }
```

```
1730            { ++
1731
1732            ALT_SOURCE -- Look for the answer elsewhere.
1733
1734            This function can look in the definition linked list or the analysis linked
1735            list, if it is determined that asking the user isn't appropriate.
1736
1737            CALLING SEQUENCE:
1738
1739            BOOLEAN_VAR := ALT_SOURCE (LINE_OBJECT_TYPE,PRIMARY,PRINUM,SECONDARY,SECNUM,AN_FLAG);
1740
1741            INPUT PARAMETERS:
1742
1743            OBJECT_TYPE
1744            PRIMARY
1745            PRINUM
1746            SECONDARY
1747            SECNUM
1748            AN_FLAG
1749
1750            IMPLICIT INPUTS:
1751
1752            none
1753
1754            OUTPUT PARAMETERS:
1755
1756            none
1757
1758            IMPLICIT OUTPUTS:
1759
1760            none
1761
1762            ROUTINES CALLED:
1763
1764            none
1765
1766            ROUTINE VALUE:
1767
1768            TRUE if the question should be asked, FALSE if it should be skipped.
1769
1770            SIGNALS:
1771
1772            none
1773
1774            SIDE EFFECTS:
1775
1776            none
1777
1778            -- }
```

```
1780        FUNCTION ALT_SOURCE (
1781                                OBJ_TYP         : LINE_OBJECT_TYPE;
1782                                PRIM            : PRIMARY_TYPE;
1783                                PRIMNUM         : INTEGER;
1784                                SECO            : SECONDARY_TYPE;
1785                                SECONUM         : INTEGER;
1786                                AN_FLAG         : BOOLEAN
1787                                ) : BOOLEAN;
1788
1789        BEGIN
1790
1791            { +
1792            Initial setup for GLOBAL_COUNT question.
1793            - }
1794            IF QTAB_OFFSET = EDF$K_GLOBAL_COUNT THEN
1795
1796                GLOBAL_SET              := FALSE;
1797
1798            { +
1799            Should the question be visible?
1800            The questions asking compression percent are never visible and should
1801            always be found in the analysis file.
1802            - }
1803            IF (
1804            ((VISIBLE_QUESTION) OR (NOT OPTIMIZING))
1805            AND
1806            ( NOT (QTAB_OFFSET IN [ EDF$K_DATA_RECORD_COMP, EDF$K_DATA_KEY_COMP,
1807                                    EDF$K_INDEX_RECORD_COMP ]))
1808            ) THEN
1809
1810            BEGIN
1811
1812                ALT_SOURCE      := TRUE;
1813
1814            END
1815
1816            ELSE IF OPTIMIZING THEN
1817
1818            BEGIN
1819
1820                { +
1821                Try to get the data from the alternate source.
1822                - }
1823                IF AN_FLAG THEN
1824
1825                    POINT_AT_ANALYSIS;
1826
1827                IF FIND_OBJECT (OBJ_TYP,PRIM,PRIMNUM,SECO,SECONUM) THEN
1828
1829                BEGIN
1830
1831                    ALT_SOURCE  := FALSE;
1832
1833                    CASE QTAB_OFFSET OF
1834
1835                        EDF$K_KEY_NAME,
1836                        EDF$K_FDL_TITLE,
```

```
1837                    EDF$K_DATA_FILE_NAME :
1838
1839                    BEGIN
1840
1841                        LIB$SCOPY_DXDX (DEF_CURRENT^.STRING,SDATA[QTAB_OFFSET]);
1842                        BDATA[QTAB_OFFSET] := TRUE;
1843
1844                    END;
1845
1846                    EDF$K_NUMBER_DUPS,
1847                    EDF$K_DESIRED_FILL,
1848                    EDF$K_CONTROL_SIZE,
1849                    EDF$K_MAX_RECORD_SIZE,
1850                    EDF$K_MEAN_RECORD_SIZE,
1851                    EDF$K_DATA_KEY_COMP,
1852                    EDF$K_DATA_RECORD_COMP,
1853                    EDF$K_CLUSTER_SIZE,
1854                    EDF$K_PROLOGUE_VERSION,
1855                    EDF$K_INDEX_RECORD_COMP :
1856
1857                    BEGIN
1858
1859                        IDATA[QTAB_OFFSET]  := DEF_CURRENT^.NUMBER;
1860                        INPUT_VALUE         := IDATA[QTAB_OFFSET];
1861
1862                    END;
1863
1864                    EDF$K_KEY_TYPE,
1865                    EDF$K_CARR_CTRL,
1866                    EDF$K_RECORD_FORMAT :
1867
1868                    BEGIN
1869
1870                        IDATA[QTAB_OFFSET]  := DEF_CURRENT^.QUALIFIER;
1871                        INPUT_VALUE         := IDATA[QTAB_OFFSET];
1872
1873                    END;
1874
1875                    EDF$K_KEY_POSITION :
1876
1877                    BEGIN
1878
1879                        IDATA[QTAB_OFFSET]                  := DEF_CURRENT^.NUMBER;
1880                        INPUT_VALUE                         := IDATA[QTAB_OFFSET];
1881                        SEGMENT_POSITION[SEGMENT_NUMBER]    := INPUT_VALUE;
1882
1883                    END;
1884
1885                    EDF$K_KEY_SIZE :
1886
1887                    BEGIN
1888
1889                        IDATA[QTAB_OFFSET]                  := DEF_CURRENT^.NUMBER;
1890                        INPUT_VALUE                         := IDATA[QTAB_OFFSET];
1891                        SEGMENT_LENGTH[SEGMENT_NUMBER]      := INPUT_VALUE;
1892
1893                    END;
```

```
1894
1895                    EDF$K_GLOBAL_COUNT :
1896
1897                    BEGIN
1898
1899                        IDATA[QTAB_OFFSET]  := DEF_CURRENT^.NUMBER;
1900                        INPUT_VALUE         := IDATA[QTAB_OFFSET];
1901                        GLOBAL_SET          := TRUE;
1902
1903                    END;    { EDF$K_GLOBAL_COUNT }
1904
1905                    EDF$K_KEY_DUPS,
1906                    EDF$K_KEY_CHANGES,
1907                    EDF$K_KEY_COMP_WANTED,
1908                    EDF$K_REC_COMP_WANTED,
1909                    EDF$K_IDX_COMP_WANTED :
1910
1911                    BEGIN
1912
1913                        BDATA[QTAB_OFFSET]  := DEF_CURRENT^.SWITCH;
1914
1915                        IF DEF_CURRENT^.SWITCH THEN
1916
1917                            INPUT_VALUE     := EDF$K_YES
1918
1919                        ELSE
1920
1921                            INPUT_VALUE     := EDF$K_NO;
1922
1923                        END;    { EDF$K_KEY_DUPS }
1924
1925                    OTHERWISE
1926
1927                        { NULL-STATEMENT } ;
1928
1929                    END;        { CASE }
1930
1931                END     { IF FOUND }
1932
1933                ELSE
1934
1935                BEGIN
1936
1937                    { +
1938                    We couldn't find it - ask the user directly.
1939                    - }
1940                    ALT_SOURCE  := TRUE;
1941
1942                    { +
1943                    Unless we're in /NOINTERACTIVE, in which case,
1944                    exit with an error.
1945                    - }
1946                    IF (
1947                    (AUTO_TUNE)
1948                    AND
1949                    (QTAB_OFFSET IN [ EDF$K_INITIAL_COUNT, EDF$K_KEY_SIZE,
1950                     EDF$K_MEAN_RECORD_SIZE ])
```

```
1951                    ) THEN
1952
1953                        LIB$STOP (EDF$_INSFANL,0,0,0);
1954
1955                    IF (QTAB_OFFSET IN [ EDF$K_DATA_RECORD_COMP, EDF$K_DATA_KEY_COMP,
1956                                         EDF$K_INDEX_RECORD_COMP ]) THEN
1957
1958                    BEGIN
1959
1960                        ALT_SOURCE              := FALSE;
1961                        RDATA[QTAB_OFFSET]      := 0.0;
1962
1963                    END;
1964
1965                END;    { IF NOT FOUND }
1966
1967                POINT_AT_DEFINITION;
1968
1969            END;        { IF FALSE (VISIBLE_QUESTION) OR (NOT OPTIMIZING) }
1970
1971        END;    { ALT_SOURCE }
```

```
1973            { ++
1974
1975            PRE_PROCESS -- Routine to setup question.
1976
1977            This function does any needed calculations before asking the question.
1978
1979            CALLING SEQUENCE:
1980
1981            status := PRE_PROCESS;
1982
1983            INPUT PARAMETERS:
1984
1985            none
1986
1987            IMPLICIT INPUTS:
1988
1989            none
1990
1991            OUTPUT PARAMETERS:
1992
1993            none
1994
1995            IMPLICIT OUTPUTS:
1996
1997            none
1998
1999            ROUTINES CALLED:
2000
2001            none
2002
2003            ROUTINE VALUE:
2004
2005            TRUE if we should continue, FALSE otherwise
2006
2007            SIGNALS:
2008
2009            none
2010
2011            SIDE EFFECTS:
2012
2013
2014            -- }
```

```
2016            FUNCTION PRE_PROCESS : BOOLEAN;
2017
2018            VAR
2019                RECORD_OVERHEAD     : INTEGER;
2020                TEMP_KEY_SIZE       : INTEGER;
2021                RESULT              : BOOLEAN;
2022
2023            BEGIN
2024
2025                { +
2026                Assume success.
2027                - }
2028                PRE_PROCESS         := TRUE;
2029
2030                CASE QTAB_OFFSET OF
2031
2032                    EDF$K_SURFACE_OPTION :
2033
2034                        IF (
2035                        (IDATA[EDF$K_ACTIVE_KEY] <> 0)
2036                        AND
2037                        ( NOT (QTAB[QTAB_OFFSET].DEFAULT IN [ EDF$K_LINE_SURFACE,
2038                                EDF$K_FILL_SURFACE, EDF$K_KEY_SURFACE ]))
2039                        ) THEN
2040
2041                            QTAB[QTAB_OFFSET].DEFAULT       := EDF$K_LINE_SURFACE;
2042
2043                    EDF$K_ADDED_COUNT_HIGH,
2044                    EDF$K_INITIAL_COUNT_HIGH :
2045
2046                        IF IDATA[EDF$K_Y_LOW] = 0 THEN
2047
2048                            DEF     := 100000
2049
2050                        ELSE
2051
2052                            DEF     := 50 * IDATA[EDF$K_Y_LOW];
2053
2054                EDF$K_KEY_LOW :
2055
2056                BEGIN
2057
2058                    { +
2059                    See how far we can go.
2060                    - }
2061                    IF (
2062                    (IDATA[EDF$K_MAX_RECORD_SIZE] = 0)
2063                    OR
2064                    (IDATA[EDF$K_MAX_RECORD_SIZE] > 255)
2065                    ) THEN
2066
2067                        MAX_KEY_SIZE    := 255
2068
2069                    ELSE
2070
2071                        MAX_KEY_SIZE    := IDATA[EDF$K_MAX_RECORD_SIZE];
2072
```

```
2073                    QTAB[QTAB_OFFSET].HIGH_BOUND        := MAX_KEY_SIZE;
2074                    QTAB[QTAB_OFFSET+1].HIGH_BOUND      := MAX_KEY_SIZE;
2075                    QTAB[QTAB_OFFSET+1].DEFAULT         := MAX_KEY_SIZE;
2076
2077                END;    { EDF$K_KEY_LOW }
2078
2079                EDF$K_DATA_KEY_COMP :
2080
2081                BEGIN
2082
2083                    PRE_PROCESS         := FALSE;
2084                    IDATA[QTAB_OFFSET]  := 0;
2085
2086                    IF (
2087                    (VDATA[EDF$K_PROLOGUE_VERSION])
2088                    AND
2089                    (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2090                    ) THEN
2091
2092                        RESULT          := ALT_SOURCE (SEC,ANALYSIS_OF_KEY,
2093                                IDATA[EDF$K_ACTIVE_KEY],DATA_KEY_COMPRESSION$,0,TRUE);
2094
2095                END;    { EDF$K_DATA_KEY_COMP }
2096
2097                EDF$K_DATA_RECORD_COMP :
2098
2099                BEGIN
2100
2101                    PRE_PROCESS         := FALSE;
2102                    IDATA[QTAB_OFFSET]  := 0;
2103
2104                    IF (
2105                    (VDATA[EDF$K_PROLOGUE_VERSION])
2106                    AND
2107                    (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2108                    ) THEN
2109
2110                        RESULT          := ALT_SOURCE (SEC,ANALYSIS_OF_KEY,
2111                                IDATA[EDF$K_ACTIVE_KEY],DATA_RECORD_COMPRESSION$,0,TRUE);
2112
2113                END;    { EDF$K_DATA_RECORD_COMP }
2114
2115                EDF$K_INDEX_RECORD_COMP :
2116
2117                BEGIN
2118
2119                    PRE_PROCESS         := FALSE;
2120                    IDATA[QTAB_OFFSET]  := 0;
2121
2122                    IF (
2123                    (VDATA[EDF$K_PROLOGUE_VERSION])
2124                    AND
2125                    (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2126                    ) THEN
2127
2128                        RESULT          := ALT_SOURCE (SEC,ANALYSIS_OF_KEY,
2129                                IDATA[EDF$K_ACTIVE_KEY],INDEX_COMPRESSION$,0,TRUE);
```

```
2130
2131                END;    { EDF$K_INDEX_RECORD_COMP }
2132
2133                EDF$K_KEY_POSITION :
2134
2135                BEGIN
2136
2137                    { +
2138                    See how far we can go.
2139                    - }
2140                    IF BDATA[EDF$K_SEGMENTED] THEN
2141
2142                    BEGIN
2143
2144                        TEMP_KEY_SIZE   := 0;
2145
2146                        FOR TEMP_INT2 := 0 TO 7 DO
2147
2148                        BEGIN
2149
2150                            IF SEGMENT_WANTED[TEMP_INT2] THEN
2151
2152                                TEMP_KEY_SIZE   := TEMP_KEY_SIZE +
2153                                                   SEGMENT_LENGTH[TEMP_INT2];
2154
2155                        END;
2156
2157                    END
2158
2159                    ELSE
2160
2161                        TEMP_KEY_SIZE   := IDATA[EDF$K_KEY_SIZE];
2162
2163                    IF IDATA[EDF$K_MAX_RECORD_SIZE] = 0 THEN
2164
2165                        MAX_KEY_POSITION        := CUR_MAX_REC - TEMP_KEY_SIZE
2166
2167                    ELSE
2168
2169                        MAX_KEY_POSITION        := IDATA[EDF$K_MAX_RECORD_SIZE]
2170                                                   - TEMP_KEY_SIZE;
2171
2172
2173                    QTAB[QTAB_OFFSET].HIGH_BOUND        := MAX_KEY_POSITION;
2174
2175                    PRE_PROCESS := ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],
2176                                               SEG_POSITION,SEGMENT_NUMBER,FALSE);
2177
2178                END;    { EDF$K_KEY_POSITION }
2179
2180                EDF$K_KEY_DIST :
2181
2182                    IF (
2183                    (IDATA[EDF$K_ACTIVE_KEY] = 0)
2184                    OR
2185                    (NOT VDATA[EDF$K_KEY_DIST])
2186                    ) THEN
```

```
2187
2188                    IF IDATA[EDF$K_ADDED_COUNT] > 0 THEN
2189
2190                        PRE_PROCESS         := TRUE
2191
2192                    ELSE
2193
2194                    BEGIN
2195
2196                        BDATA[QTAB_OFFSET]  := TRUE;
2197                        PRE_PROCESS         := FALSE;
2198
2199                    END
2200
2201                ELSE
2202
2203                    PRE_PROCESS             := FALSE;
2204
2205            EDF$K_KEY_CHANGES :
2206
2207                IF IDATA[EDF$K_ACTIVE_KEY] <> 0 THEN
2208
2209                    PRE_PROCESS     := ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],
2210                                       CHANGES,0,FALSE)
2211
2212                ELSE
2213
2214                BEGIN
2215
2216                    PRE_PROCESS             := FALSE;
2217                    BDATA[QTAB_OFFSET]      := FALSE;
2218
2219                END;
2220
2221            EDF$K_KEY_DUPS :
2222
2223            BEGIN
2224
2225                IF IDATA[EDF$K_ACTIVE_KEY] = 0 THEN
2226
2227                    QTAB[QTAB_OFFSET].DEFAULT       := EDF$K_NO
2228
2229                ELSE
2230
2231                    QTAB[QTAB_OFFSET].DEFAULT       := EDF$K_YES;
2232
2233                PRE_PROCESS := ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],
2234                        DUPLICATES,0,FALSE);
2235
2236            END;    { EDF$K_KEY_DUPS }
2237
2238            EDF$K_NUMBER_KEYS :
2239
2240                PRE_PROCESS := (
2241                                    (VISIBLE_QUESTION)
2242                                    OR
2243                                    (NOT NUMBER_KEYS_SET)
```

```
2244                                    );
2245
2246                    EDF$K_CLUSTER_SIZE :
2247
2248                    BEGIN
2249
2250                        { +
2251                        This question shouldn't be asked for alternate keys,
2252                        unless it hasn't been asked yet. Or unless we're optimizing.
2253                        - }
2254                        IF OPTIMIZING THEN
2255
2256                            PRE_PROCESS     := ALT_SOURCE (SEC,FILE$,0,CLUSTER_SIZE,0,TRUE)
2257
2258                        ELSE
2259
2260                            PRE_PROCESS     := (
2261                                    (IDATA[EDF$K_ACTIVE_KEY] = 0)
2262                                    OR
2263                                    (NOT VDATA[EDF$K_ADDED_COUNT])
2264                                    );
2265
2266                    END;    { EDF$K_CLUSTER_SIZE }
2267
2268                    EDF$K_KEY_COMP_WANTED :
2269
2270                    BEGIN
2271
2272                        PRE_PROCESS         := FALSE;
2273                        BDATA[QTAB_OFFSET]  := FALSE;
2274
2275                        IF (
2276                        (VDATA[EDF$K_PROLOGUE_VERSION])
2277                        AND
2278                        (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2279                        AND
2280                        (IDATA[EDF$K_KEY_TYPE] = FDL$C_STG)
2281                        ) THEN
2282
2283                            PRE_PROCESS := ALT_SOURCE (SEC,KEY,
2284                                    IDATA[EDF$K_ACTIVE_KEY],DATA_KEY_COMPRESSION,0,FALSE);
2285
2286                    END;    { EDF$K_DATA_RECORD_WANTED }
2287
2288                    EDF$K_REC_COMP_WANTED :
2289
2290                    BEGIN
2291
2292                        PRE_PROCESS         := FALSE;
2293                        BDATA[QTAB_OFFSET]  := FALSE;
2294
2295                        IF (
2296                        (VDATA[EDF$K_PROLOGUE_VERSION])
2297                        AND
2298                        (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2299                        AND
2300                        (IDATA[EDF$K_KEY_TYPE] = FDL$C_STG)
```

```
2301                        AND
2302                        (IDATA[EDF$K_ACTIVE_KEY] = 0)
2303                        ) THEN
2304
2305                            PRE_PROCESS := ALT_SOURCE (SEC,KEY,
2306                                 IDATA[EDF$K_ACTIVE_KEY],DATA_RECORD_COMPRESSION,0,FALSE);
2307
2308                    END;    { EDF$K_REC_COMP_WANTED }
2309
2310                    EDF$K_IDX_COMP_WANTED :
2311
2312                    BEGIN
2313
2314                        PRE_PROCESS          := FALSE;
2315                        BDATA[QTAB_OFFSET]   := FALSE;
2316
2317                        IF (
2318                        (VDATA[EDF$K_PROLOGUE_VERSION])
2319                        AND
2320                        (IDATA[EDF$K_PROLOGUE_VERSION] > 2)
2321                        AND
2322                        (IDATA[EDF$K_KEY_TYPE] = FDL$C_STG)
2323                        ) THEN
2324
2325                            PRE_PROCESS := ALT_SOURCE (SEC,KEY,
2326                                    IDATA[EDF$K_ACTIVE_KEY],INDEX_COMPRESSION,0,FALSE);
2327
2328                    END;    { EDF$K_IDX_COMP_WANTED }
2329
2330                    EDF$K_NUMBER_DUPS :
2331
2332                    BEGIN
2333
2334                        PRE_PROCESS                 := FALSE;
2335                        IDATA[EDF$K_NUMBER_DUPS]    := 0;
2336
2337                        RESULT      := ALT_SOURCE (SEC,ANALYSIS_OF_KEY,
2338                                IDATA[EDF$K_ACTIVE_KEY],DUPLICATES_PER_SIDR,0,TRUE);
2339
2340                    END;
2341
2342                    EDF$K_SEGMENTED :
2343
2344                    BEGIN
2345
2346                        FOR TEMP_INT2 := 0 TO 7 DO
2347
2348                        BEGIN
2349
2350                            SEGMENT_WANTED[TEMP_INT2]       := FALSE;
2351                            SEGMENT_POSITION[TEMP_INT2]     := 0;
2352                            SEGMENT_LENGTH[TEMP_INT2]       := 0;
2353
2354                        END;
2355
2356                        BDATA[QTAB_OFFSET]  := FALSE;
2357
```

```
2358                    IF IDATA[EDF$K_KEY_TYPE] <> FDL$C_STG THEN
2359
2360                        PRE_PROCESS     := FALSE;
2361
2362                END;
2363
2364                EDF$K_GLOBAL_WANTED :
2365
2366                BEGIN
2367
2368                    { +
2369                    Set global_wanted by the presence or absence of a
2370                    global_buffer_count secondary in the list, when optimizing.
2371                    THIS DOESN'T USE ALT_SOURCE BECAUSE IT DOESN'T GET THE
2372                    CONTENTS OF THE SECONDARY IN THE LIST, BUT DECIDES ON THE
2373                    BASIS OF ITS EXISTENCE.
2374                    - }
2375                    IF OPTIMIZING THEN
2376
2377                    BEGIN
2378
2379                        IF FIND_OBJECT (SEC,FILE$,0,GLOBAL_BUFFER_COUNT,0) THEN
2380
2381                            INPUT_VALUE         := EDF$K_YES
2382
2383                        ELSE
2384
2385                            INPUT_VALUE         := EDF$K_NO;
2386
2387                        PRE_PROCESS             := FALSE;
2388
2389                    END;
2390
2391                END;    { EDF$K_GLOBAL_WANTED }
2392
2393                EDF$K_GLOBAL_COUNT :
2394
2395                BEGIN
2396
2397                    PRE_PROCESS := FALSE;
2398                    RESULT      := ALT_SOURCE (SEC,FILE$,0,GLOBAL_BUFFER_COUNT,0,FALSE);
2399
2400                END;
2401
2402                EDF$K_INITIAL_COUNT :
2403
2404                BEGIN
2405
2406                    { +
2407                    This question shouldn't be asked for alternate keys,
2408                    unless it hasn't been asked yet.
2409                    - }
2410                    PRE_PROCESS := (
2411                            (IDATA[EDF$K_ACTIVE_KEY] = 0)
2412                            OR
2413                            (NOT VDATA[QTAB_OFFSET])
2414                            );
```

```
2415
2416                    IF OPTIMIZING THEN
2417
2418                    BEGIN
2419
2420                        POINT_AT_ANALYSIS;
2421
2422                        IF FIND_OBJECT (SEC,ANALYSIS_OF_KEY,0,DATA_RECORD_COUNT,0) THEN
2423
2424                            OLD_COUNT   := DEF_CURRENT^.NUMBER
2425
2426                        ELSE
2427
2428                            OLD_COUNT   := 0;
2429
2430                        POINT_AT_DEFINITION;
2431
2432                        QTAB[QTAB_OFFSET].DEFAULT_OK    := TRUE;
2433                        QTAB[QTAB_OFFSET].DEFAULT       := OLD_COUNT;
2434
2435                    END
2436
2437                    ELSE
2438
2439                        QTAB[QTAB_OFFSET].DEFAULT_OK    := FALSE;
2440
2441                END;    { EDF$K_INITIAL_COUNT }
2442
2443                EDF$K_ASCENDING_LOAD :
2444
2445                    { +
2446                    This question shouldn't be asked if we're doing a conv/fast,
2447                    or for alternate keys, unless it hasn't been asked yet.
2448                    - }
2449                    PRE_PROCESS := (
2450                            (IDATA[EDF$K_INITIAL_COUNT] > 0)
2451                        AND
2452                            (IDATA[EDF$K_LOAD_METHOD] <> EDF$K_FAST_CONVERT)
2453                        AND
2454                            (
2455                            (IDATA[EDF$K_ACTIVE_KEY] = 0)
2456                            OR
2457                            (NOT VDATA[EDF$K_ASCENDING_LOAD])
2458                            )
2459                        );
2460
2461                EDF$K_ADDED_COUNT :
2462
2463                    { +
2464                    This question shouldn't be asked for alternate keys,
2465                    unless it hasn't been asked yet.
2466                    - }
2467                    PRE_PROCESS := (
2468                            (IDATA[EDF$K_ACTIVE_KEY] = 0)
2469                            OR
2470                            (NOT VDATA[EDF$K_ADDED_COUNT])
2471                            );
```

```
2472
2473            EDF$K_LOAD_METHOD :
2474
2475            BEGIN
2476
2477                IF IDATA[EDF$K_INITIAL_COUNT] > 0 THEN
2478
2479                    RESULT  := (
2480                            (IDATA[EDF$K_ACTIVE_KEY] = 0)
2481                            OR
2482                            (NOT VDATA[QTAB_OFFSET])
2483                            )
2484
2485                ELSE
2486
2487                BEGIN
2488
2489                    { +
2490                    If we have no initial load, default it to rms puts
2491                    - }
2492                    IDATA[QTAB_OFFSET]      := EDF$K_RMS_PUTS;
2493                    RESULT                  := FALSE;
2494
2495                END;        { IF FALSE IDATA[EDF$K_INITIAL_COUNT] > 0 }
2496
2497                IF NOT RESULT THEN
2498
2499                    INPUT_VALUE             := IDATA[QTAB_OFFSET];
2500
2501                PRE_PROCESS                 := RESULT;
2502
2503            END;    { EDF$K_LOAD_METHOD }
2504
2505            EDF$K_BLOCK_SPAN :
2506
2507            BEGIN
2508
2509                IF IDATA[EDF$K_SCRIPT_OPTION] = EDF$K_SEQ_DESIGN_FDL THEN
2510
2511                    PRE_PROCESS     := TRUE
2512
2513                ELSE
2514
2515                BEGIN
2516
2517                    PRE_PROCESS             := FALSE;
2518                    INPUT_VALUE             := EDF$K_YES;
2519
2520                END;
2521
2522            END;    { EDF$K_BLOCK_SPAN }
2523
2524            EDF$K_DATA_FILE_NAME :
2525
2526            BEGIN
2527
2528                MAX_STRING_ANSWER_LENGTH    := 126;
```

```
2529                    BDATA[QTAB_OFFSET]          := FALSE;
2530
2531                    IF OPTIMIZING THEN
2532
2533                        PRE_PROCESS     :=
2534                            ALT_SOURCE (SEC,FILE$,0,NAME,0,FALSE);
2535
2536                END;    { EDF$K_DATA_FILE_NAME }
2537
2538                EDF$K_FDL_TITLE :
2539
2540                BEGIN
2541
2542                    MAX_STRING_ANSWER_LENGTH    := 126;
2543                    BDATA[QTAB_OFFSET]          := FALSE;
2544
2545                    IF OPTIMIZING THEN
2546
2547                        PRE_PROCESS     :=
2548                            ALT_SOURCE (PRI,TITLE,0,DUMMY_SECONDARY$,0,FALSE);
2549
2550                END;    { EDF$K_FDL_TITLE }
2551
2552                EDF$K_KEY_NAME :
2553
2554                BEGIN
2555
2556                    MAX_STRING_ANSWER_LENGTH    := 32;
2557                    BDATA[QTAB_OFFSET]          := FALSE;
2558
2559                    IF OPTIMIZING THEN
2560
2561                        PRE_PROCESS     :=
2562                            ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],NAME$,0,FALSE);
2563
2564                END;
2565
2566                EDF$K_ANALYSIS,
2567                EDF$K_OUTPUT :
2568
2569                BEGIN
2570
2571                    MAX_STRING_ANSWER_LENGTH    := 126;
2572                    BDATA[QTAB_OFFSET]          := FALSE;
2573
2574                    IF OPTIMIZING THEN
2575
2576                        PRE_PROCESS             := FALSE;
2577
2578                END;
2579
2580                EDF$K_ASCENDING_ADDED :
2581
2582                    { +
2583                    This question shouldn't be asked for alternate keys,
2584                    unless it hasn't been asked yet.
2585                    - }
```

```
2586                    PRE_PROCESS := (
2587                            (IDATA[EDF$K_ADDED_COUNT] > 0)
2588                            AND
2589                                ((IDATA[EDF$K_ACTIVE_KEY] = 0)
2590                                 OR
2591                                (NOT VDATA[EDF$K_ASCENDING_ADDED]))
2592                            );
2593
2594                EDF$K_BLOCKS_IN_BUCKET :
2595
2596                BEGIN
2597
2598                    QTAB[QTAB_OFFSET].DEFAULT   := BUCKET_DEFAULT;
2599
2600                    { +
2601                    Calculate the bucket overhead.
2602                    THIS QUESTION IS ONLY FOR INDEXED_DESIGN.
2603                    - }
2604                    BUCKET_OVERHEAD     := CALC_BUC_OVERHEAD(0);
2605
2606                    { +
2607                    See what the smallest allowable bucketsize is.
2608                    - }
2609                    IF IDATA[EDF$K_ACTIVE_KEY] = 0 THEN
2610
2611                        ENTRY_SIZE      := IDATA[EDF$K_MAX_RECORD_SIZE]
2612
2613                    ELSE
2614
2615                    BEGIN
2616
2617                        IF BDATA[EDF$K_SEGMENTED] THEN
2618
2619                        BEGIN
2620
2621                            ENTRY_SIZE  := 0;
2622
2623                            FOR TEMP_INT2 := 0 TO 7 DO
2624
2625                            BEGIN
2626
2627                                IF SEGMENT_WANTED[TEMP_INT2] THEN
2628
2629                                    ENTRY_SIZE  := ENTRY_SIZE +
2630                                                   SEGMENT_LENGTH[TEMP_INT2];
2631
2632                            END;
2633
2634                        END
2635
2636                        ELSE
2637
2638                            ENTRY_SIZE  := IDATA[EDF$K_KEY_SIZE];
2639
2640                END;
2641
2642                RECORD_OVERHEAD     := CALC_REC_OVERHEAD(0);
```

```
2643
2644            MIN_BUCKET  := (ENTRY_SIZE + BUCKET_OVERHEAD + RECORD_OVERHEAD)
2645                           DIV 512;
2646
2647            IF (
2648            (((ENTRY_SIZE + BUCKET_OVERHEAD + RECORD_OVERHEAD) MOD 512) <> 0)
2649            OR
2650            (MIN_BUCKET = 0)
2651            ) THEN
2652
2653                MIN_BUCKET      := MIN_BUCKET + 1;
2654
2655            QTAB[QTAB_OFFSET].LOW_BOUND := MIN_BUCKET;
2656
2657            IF QTAB[QTAB_OFFSET].DEFAULT < QTAB[QTAB_OFFSET].LOW_BOUND THEN
2658
2659                QTAB[QTAB_OFFSET].DEFAULT       := QTAB[QTAB_OFFSET].LOW_BOUND;
2660
2661        END;    { EDF$K_BLOCKS_IN_BUCKET }
2662
2663        EDF$K_KEY_SIZE :
2664
2665        BEGIN
2666
2667            { +
2668            Check according to key type.
2669            - }
2670            CASE IDATA[EDF$K_KEY_TYPE] OF
2671
2672                FDL$C_BN2, FDL$C_IN2 :
2673
2674                BEGIN
2675
2676                    MAX_KEY_SIZE        := 2;
2677                    MIN_KEY_SIZE        := 2;
2678
2679                END;
2680
2681                FDL$C_BN4, FDL$C_IN4 :
2682
2683                BEGIN
2684
2685                    MAX_KEY_SIZE        := 4;
2686                    MIN_KEY_SIZE        := 4;
2687
2688                END;
2689
2690                FDL$C_BN8, FDL$C_IN8 :
2691
2692                BEGIN
2693
2694                    MAX_KEY_SIZE        := 8;
2695                    MIN_KEY_SIZE        := 8;
2696
2697                END;
2698
2699                FDL$C_PAC :
```

```
2700
2701                    BEGIN
2702
2703                        MAX_KEY_SIZE        := 16;
2704                        MIN_KEY_SIZE        := 1;
2705
2706                    END;
2707
2708                    FDL$C_STG :
2709
2710                    BEGIN
2711
2712                        MAX_KEY_SIZE        := 255;
2713                        MIN_KEY_SIZE        := 1;
2714
2715                    END;
2716
2717                OTHERWISE
2718
2719                    { NULL-STATEMENT } ;
2720
2721                END;        { CASE }
2722
2723                { +
2724                See how far we can go.
2725                - }
2726                IF (
2727                (IDATA[EDF$K_MAX_RECORD_SIZE] <> 0)
2728                AND
2729                (IDATA[EDF$K_MAX_RECORD_SIZE] < MAX_KEY_SIZE)
2730                ) THEN
2731
2732                    MAX_KEY_SIZE    := IDATA[EDF$K_MAX_RECORD_SIZE];
2733
2734                QTAB[QTAB_OFFSET].LOW_BOUND         := MIN_KEY_SIZE;
2735                QTAB[QTAB_OFFSET].HIGH_BOUND        := MAX_KEY_SIZE;
2736
2737                RESULT      :=
2738                    ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],SEG_LENGTH,
2739                                SEGMENT_NUMBER,FALSE);
2740                PRE_PROCESS := RESULT;
2741
2742                IF (
2743                (RESULT)
2744                AND
2745                (QTAB[QTAB_OFFSET].LOW_BOUND = QTAB[QTAB_OFFSET].HIGH_BOUND)
2746                ) THEN
2747
2748                BEGIN
2749
2750                    PRE_PROCESS                     := FALSE;
2751                    INPUT_VALUE                     := QTAB[QTAB_OFFSET].LOW_BOUND;
2752                    IDATA[QTAB_OFFSET]              := INPUT_VALUE;
2753                    SEGMENT_LENGTH[SEGMENT_NUMBER]  := INPUT_VALUE;
2754
2755                END;
2756
```

```
2757                END;    { EDF$K_KEY_SIZE }
2758
2759                EDF$K_PROLOGUE_VERSION :
2760
2761                BEGIN
2762
2763                    IF (
2764                    (IDATA[EDF$K_ACTIVE_KEY] = 0)
2765                    OR
2766                    (NOT VDATA[EDF$K_PROLOGUE_VERSION])
2767                    ) THEN
2768
2769                        PRE_PROCESS     :=
2770                        ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],PROLOGUE,0,FALSE)
2771
2772                    ELSE
2773
2774                        PRE_PROCESS     := FALSE;
2775
2776                END;
2777
2778                EDF$K_ACTIVE_KEY :
2779
2780                BEGIN
2781
2782                    { +
2783                    Find out the range of existing keys (assume contiguous).
2784                    - }
2785                    SCAN_DEFINITION (TRUE);
2786
2787                    QTAB[QTAB_OFFSET].LOW_BOUND         := LOW_KEY;
2788                    QTAB[QTAB_OFFSET].HIGH_BOUND        := HIGH_KEY;
2789
2790                    IF (
2791                    (QTAB[QTAB_OFFSET].LOW_BOUND = QTAB[QTAB_OFFSET].HIGH_BOUND)
2792                    ) THEN
2793
2794                    BEGIN
2795
2796                        PRE_PROCESS     := FALSE;
2797                        INPUT_VALUE     := QTAB[QTAB_OFFSET].LOW_BOUND;
2798
2799                    END;
2800
2801                END;    { EDF$K_ACTIVE_KEY }
2802
2803                EDF$K_CARR_CTRL :
2804
2805                BEGIN
2806
2807                    { +
2808                    Don't actually ask the question if the user is optimizing a key,
2809                    or if it's an alternate key - unless it hasn't been asked yet.
2810                    - }
2811                    IF (
2812                    (IDATA[EDF$K_ACTIVE_KEY] = 0)
2813                    OR
```

```
2814                (NOT VDATA[EDF$K_CARR_CTRL])
2815                ) THEN
2816
2817                    PRE_PROCESS     :=
2818                            ALT_SOURCE (SEC,RECORD$,0,CARRIAGE_CONTROL,0,FALSE)
2819
2820                ELSE
2821
2822                    PRE_PROCESS     := FALSE;
2823
2824            END;    { EDF$K_CARR_CTRL }
2825
2826            EDF$K_CONTROL_SIZE :
2827
2828            BEGIN
2829
2830                { +
2831                The fixed portion of a record can't be larger than the record.
2832                - }
2833                IF IDATA[EDF$K_MEAN_RECORD_SIZE] < 256 THEN
2834
2835                    CUR_MAX_FIXED   := IDATA[EDF$K_MEAN_RECORD_SIZE]
2836
2837                ELSE
2838
2839                    CUR_MAX_FIXED   := 255;
2840
2841                QTAB[QTAB_OFFSET].HIGH_BOUND        := CUR_MAX_FIXED;
2842
2843                PRE_PROCESS := ALT_SOURCE (SEC,RECORD$,0,CONTROL_FIELD_SIZE,0,FALSE);
2844
2845            END;            { EDF$K_CONTROL_SIZE }
2846
2847            EDF$K_KEY_TYPE :
2848
2849            BEGIN
2850
2851                PRE_PROCESS :=
2852                    ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],SEG_TYPE,7,FALSE);
2853
2854            END;            { EDF$K_KEY_TYPE }
2855
2856            EDF$K_DESIRED_FILL :
2857
2858            BEGIN
2859
2860                { +
2861                Fill doesn't mean anything if we don't have any records.
2862                - }
2863                IF IDATA[EDF$K_INITIAL_COUNT] > 0 THEN
2864
2865                BEGIN
2866
2867                    PRE_PROCESS     :=
2868                            ALT_SOURCE (SEC,KEY,IDATA[EDF$K_ACTIVE_KEY],
2869                            DATA_FILL,0,FALSE)
2870
```

```
2871                    END         { IF TRUE IDATA[EDF$K_INITIAL_COUNT > 0 }
2872
2873                    ELSE
2874
2875                    BEGIN
2876
2877                        IDATA[EDF$K_DESIRED_FILL]       := 100;
2878                        IDATA[EDF$K_FDL_FIL[]           := 100;
2879                        PRE_PROCESS                     := FALSE;
2880
2881                    END;
2882
2883                END;    { EDF$K_DESIRED_FILL }
2884
2885                EDF$K_MAX_RECORD_SIZE :
2886
2887                BEGIN
2888
2889                    { +
2890                    Because mean_record_size includes fixed control area, and maximum
2891                    record size doesn't, it's possible to get a mean that's larger
2892                    than the max. Don't get confused by it.
2893                    - }
2894                    IF CUR_MAX_REC < IDATA[EDF$K_MEAN_RECORD_SIZE] THEN
2895
2896                        LOWMAX  := CUR_MAX_REC
2897
2898                    ELSE
2899
2900                        LOWMAX  := IDATA[EDF$K_MEAN_RECORD_SIZE];
2901
2902                    QTAB[QTAB_OFFSET].LOW_BOUND         := LOWMAX;
2903                    QTAB[QTAB_OFFSET].HIGH_BOUND        := CUR_MAX_REC;
2904
2905                    PRE_PROCESS := ALT_SOURCE (SEC,RECORD$,0,SIZE,0,FALSE);
2906
2907                END;    { EDF$K_MAX_RECORD_SIZE }
2908
2909                EDF$K_SIZE_LOW :
2910
2911                            CUR_MAX_REC := (BKT$C_MAXBKTSIZ * 512) -
2912                                    (CALC_BUC_OVERHEAD(0) + CALC_REC_OVERHEAD(0));
2913
2914                EDF$K_MEAN_RECORD_SIZE :
2915
2916                BEGIN
2917
2918                    { +
2919                    Setup the max allowable record size.
2920                    - }
2921                    CASE IDATA[EDF$K_SCRIPT_OPTION] OF
2922
2923                        EDF$K_ADD_KEY_FDL,
2924                        EDF$K_DELETE_KEY_FDL,
2925                        EDF$K_IDX_DESIGN_FDL,
2926                        EDF$K_REDESIGN_FDL,
2927                        EDF$K_OPTIMIZE_FDL :
```

```
2928                            CUR_MAX_REC := (BKT$C_MAXBKTSIZ * 512) -
2929                                    (CALC_BUC_OVERHEAD(0) + CALC_REC_OVERHEAD(0));
2930                        EDF$K_SEQ_DESIGN_FDL :  CUR_MAX_REC     := BIGGEST_SEQ_FIXED;
2931                        EDF$K_REL_DESIGN_FDL :  IF VARIABLE_RECORDS THEN
2932
2933                                                    CUR_MAX_REC := BIGGEST_REL_VAR
2934
2935                                                ELSE
2936
2937                                                    CUR_MAX_REC := BIGGEST_REL_FIXED;
2938
2939                    OTHERWISE
2940
2941                        { NULL-STATEMENT } ;
2942
2943                    END;        { CASE }
2944
2945                    IF (
2946                    (IDATA[EDF$K_SCRIPT_OPTION] = EDF$K_SEQ_DESIGN_FDL)
2947                    AND
2948                    (NOT BDATA[EDF$K_BLOCK_SPAN])
2949                    ) THEN
2950
2951                        IF VARIABLE_RECORDS THEN
2952
2953                            CUR_MAX_REC := 510
2954
2955                        ELSE
2956
2957                            CUR_MAX_REC := 512;
2958
2959                    QTAB[QTAB_OFFSET].HIGH_BOUND        := CUR_MAX_REC;
2960
2961                        IF VARIABLE_RECORDS THEN
2962
2963                            PRE_PROCESS     :=
2964                                ALT_SOURCE (SEC,ANALYSIS_OF_KEY,0,MEAN_DATA_LENGTH,0,TRUE)
2965
2966                        ELSE
2967
2968                            PRE_PROCESS     := ALT_SOURCE (SEC,RECORD$,0,SIZE,0,FALSE);
2969
2970                    END;    { EDF$K_MEAN_RECORD_SIZE }
2971
2972                    EDF$K_RECORD_FORMAT :
2973
2974                    BEGIN
2975
2976                        { +
2977                        This question shouldn't be asked for alternate keys,
2978                        unless it hasn't been asked before.
2979                        - }
2980                        IF (
2981                        (NOT ISAM_ORG)
2982                        OR
2983                        (IDATA[EDF$K_ACTIVE_KEY] = 0)
2984                        OR
```

```
2985                (NOT VDATA[EDF$K_RECORD_FORMAT])
2986                ) THEN
2987
2988                    PRE_PROCESS     := ALT_SOURCE (SEC,RECORD$,0,FORMAT,0,FALSE)
2989
2990                ELSE
2991
2992                    PRE_PROCESS     := FALSE;
2993
2994            END;    { EDF$K_RECORD_FORMAT }
2995
2996            EDF$K_DESIGN_CYCLE :
2997
2998            BEGIN
2999
3000                IF NOT AUTO_TUNE THEN
3001
3002                BEGIN
3003
3004                    { +
3005                    Display the current value of the file parameters.
3006                    - }
3007                    CLEAR (LOWER_AREA);
3008
3009                    { +
3010                    Special support for the VT125. Turn on graphics mode, setup text.
3011                    - }
3012                    IF REGIS THEN
3013
3014                    BEGIN
3015
3016                        WRITE (''(27)'Pp;');
3017
3018                        IF IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_LINE_SURFACE THEN
3019
3020                            WRITE (
3021        'P[27,285];T(W(I3))'' Key:    Good   '';T(W(I2))''Fair   '';T(W(I1))''Poor'';');
3022
3023                        WRITELN ('P[27,320];T(W(I3))''');
3024
3025                    END;    { IF REGIS }
3026
3027                    WRITE (LOW_SHIFT,' ');
3028
3029                    WRITE ('PV-Prolog Version       ',IDATA[EDF$K_PROLOGUE_VERSION]:1,' ');
3030                    WRITE ('KT-Key',
3031                    IDATA[EDF$K_ACTIVE_KEY]:3,
3032                    ' Type    ');
3033
3034                    CASE IDATA[EDF$K_KEY_TYPE] OF
3035
3036                        FDL$C_BN2 :         WRITE ('   Bin2 ');
3037                        FDL$C_BN4 :         WRITE ('   Bin4 ');
3038                        FDL$C_BN8 :         WRITE ('   Bin8 ');
3039                        FDL$C_PAC :         WRITE ('Decimal ');
3040                        FDL$C_IN2 :         WRITE ('   Int2 ');
3041                        FDL$C_IN4 :         WRITE ('   Int4 ');
```

```
3042                        FDL$C_IN8 :         WRITE ('   Int8 ');
3043                        FDL$C_STG :         WRITE ('  String ');
3044
3045                    OTHERWISE
3046
3047                        { NULL-STATEMENT } ;
3048
3049                    END;    { CASE }
3050
3051                    WRITE ('EM-Emphasis  ');
3052
3053                    IF IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_LINE_SURFACE THEN
3054
3055                        WRITE ('     ');
3056
3057                    IF IDATA[EDF$K_BUCKET_WEIGHT] = EDF$K_SMALLER_BUFFERS THEN
3058
3059                        WRITE ('Smaller')
3060
3061                    ELSE
3062
3063                        WRITE ('Flatter');
3064
3065                    IF IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE THEN
3066
3067                    BEGIN
3068
3069                        BUCKET_DEFAULT      := NATURAL_DEPTH;
3070                        WRITE (' (',BUCKET_DEFAULT:2,')');
3071
3072                    END;
3073
3074                    WRITELN;
3075                    WRITE (LOW_SHIFT,' ');
3076
3077                    WRITE ('DK-Dup Key',
3078                    IDATA[EDF$K_ACTIVE_KEY]:3,
3079                    ' Values  ');
3080
3081                    IF BDATA[EDF$K_KEY_DUPS] THEN
3082
3083                        WRITE ('Yes ')
3084
3085                    ELSE
3086
3087                        WRITE (' No ');
3088
3089                    IF (IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE)
3090                        OR (IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_KEY_SURFACE) THEN
3091
3092                        WRITE ('KL-Key',IDATA[EDF$K_ACTIVE_KEY]:3,
3093                        ' Length      ',IDATA[EDF$K_KEY_SIZE]:3,' ');
3094
3095                    WRITE ('KP-Key',IDATA[EDF$K_ACTIVE_KEY]:3,
3096                    ' Position  ',IDATA[EDF$K_KEY_POSITION]:5,' ');
3097
3098                    WRITELN;
```

```
3099                    WRITE (LOW_SHIFT,' ');
3100
3101                    IF IDATA[EDF$K_PROLOGUE_VERSION] > 2 THEN
3102
3103                    BEGIN
3104
3105                        WRITE ('RC-Data Record Comp  ',
3106                        TRUNC(RDATA[EDF$K_DATA_RECORD_COMP]*100.0):3,'% ');
3107                        WRITE ('KC-Data Key Comp     ',
3108                        TRUNC(RDATA[EDF$K_DATA_KEY_COMP]*100.0):3,'% ');
3109                        WRITE ('IC-Index Record Comp ',
3110                        TRUNC(RDATA[EDF$K_INDEX_RECORD_COMP]*100.0):3,'% ');
3111
3112                        WRITELN;
3113                        WRITE (LOW_SHIFT,' ');
3114
3115                    END;    { IF IDATA[EDF$K_PROLOGUE_VERSION] > 2 }
3116
3117                    IF (IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE)
3118                        OR (IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_FILL_SURFACE) THEN
3119
3120                        WRITE ('BF-Bucket Fill       ',IDATA[EDF$K_DESIRED_FILL]:3,'% ');
3121
3122                    WRITE ('RF-Record Format ');
3123
3124                    IF VARIABLE_RECORDS THEN
3125
3126                        WRITE ('Variable ')
3127
3128                    ELSE
3129
3130                        WRITE ('   Fixed ');
3131
3132                    IF (IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE)
3133                        OR (IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_SIZE_SURFACE) THEN
3134
3135                    BEGIN
3136
3137                        WRITE ('RS-');
3138
3139                        IF VARIABLE_RECORDS THEN
3140
3141                            WRITE ('Mean Record Size ')
3142
3143                        ELSE
3144
3145                            WRITE ('Record Size      ');
3146
3147                        WRITE (IDATA[EDF$K_MEAN_RECORD_SIZE]:5,' ');
3148
3149                    END;
3150
3151
3152                    WRITELN;
3153                    WRITE (LOW_SHIFT,' ');
3154
3155                    WRITE ('LM-Load Method ');
```

```
3156
3157                    CASE IDATA[EDF$K_LOAD_METHOD] OF
3158
3159                        EDF$K_FAST_CONVERT :                WRITE (' Fast_Conv ');
3160                        EDF$K_NOFAST_CONVERT :              WRITE ('NoFast_Con ');
3161                        EDF$K_RMS_PUTS :            WRITE ('  RMS_Puts ');
3162
3163                    OTHERWISE
3164
3165                        { NULL-STATEMENT } ;
3166
3167                    END;    { CASE }
3168
3169                    IF (IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE)
3170                        OR (IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_INIT_SURFACE) THEN
3171
3172                        WRITE ('IL-Initial Load ',IDATA[EDF$K_INITIAL_COUNT]:9,' ');
3173
3174                    IF (IDATA[EDF$K_SURFACE_OPTION] = EDF$K_LINE_SURFACE)
3175                        OR (IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_ADDED_SURFACE) THEN
3176
3177                        WRITE ('AR-Added Records',IDATA[EDF$K_ADDED_COUNT]:9,' ');
3178
3179                    { +
3180                    Done with display, now turn Graphics Mode off.
3181                    - }
3182                    IF REGIS THEN
3183
3184                        WRITELN (''';'(27)'\',CRLF,CRLF);
3185
3186                    WRITELN;
3187
3188                    { +
3189                    Compensate for absent compression line.
3190                    - }
3191                    IF IDATA[EDF$K_PROLOGUE_VERSION] < 3 THEN
3192
3193                        WRITELN;
3194
3195                END;        { IF NOT AUTO_TUNE }
3196
3197            END;    { EDF$K_DESIGN_CYCLE }
3198
3199        OTHERWISE
3200
3201            { NULL-STATEMENT } ;
3202
3203        END;        { CASE }
3204
3205    END;    { PRE_PROCESS }
```

```
3207            { ++
3208
3209            VERIFY_PROCESS -- Routine to check answer during questioning.
3210
3211            This function makes sure the user is giving good answers.
3212
3213            CALLING SEQUENCE:
3214
3215            status := VERIFY_PROCESS;
3216
3217            INPUT PARAMETERS:
3218
3219            none
3220
3221            IMPLICIT INPUTS:
3222
3223            none
3224
3225            OUTPUT PARAMETERS:
3226
3227            none
3228
3229            IMPLICIT OUTPUTS:
3230
3231            none
3232
3233            ROUTINES CALLED:
3234
3235            none
3236
3237            ROUTINE VALUE:
3238
3239            TRUE if we should continue, FALSE otherwise
3240
3241            SIGNALS:
3242
3243            none
3244
3245            SIDE EFFECTS:
3246
3247
3248            -- }
```

```
3250            FUNCTION VERIFY_PROCESS : BOOLEAN;
3251
3252            BEGIN
3253
3254                { +
3255                Assume success.
3256                - }
3257                VERIFY_PROCESS      := TRUE;
3258
3259                CASE QTAB_OFFSET OF
3260
3261                    EDF$K_KEY_NAME,
3262                    EDF$K_FDL_TITLE,
3263                    EDF$K_ANALYSIS,
3264                    EDF$K_OUTPUT,
3265                    EDF$K_DATA_FILE_NAME :
3266
3267                    BEGIN
3268
3269                        IF SDATA[QTAB_OFFSET].DSC$W_LENGTH = 0 THEN
3270
3271                        BEGIN
3272
3273                            BDATA[QTAB_OFFSET]      := FALSE;
3274
3275                        END
3276
3277                        ELSE
3278
3279                        BEGIN
3280
3281                            IF (
3282                            SDATA[QTAB_OFFSET].DSC$W_LENGTH > MAX_STRING_ANSWER_LENGTH
3283                            ) THEN
3284
3285                            BEGIN
3286
3287                                IF OPTIMIZING THEN
3288
3289                                BEGIN
3290
3291                                    SDATA[QTAB_OFFSET].DSC$W_LENGTH :=
3292                                                        MAX_STRING_ANSWER_LENGTH;
3293                                    BDATA[QTAB_OFFSET]              := TRUE;
3294
3295                                END
3296
3297                                ELSE
3298
3299                                BEGIN
3300
3301                                    STR$FREE1_DX (SDATA[QTAB_OFFSET]);
3302                                    VERIFY_PROCESS  := FALSE;
3303
3304                                END;
3305
3306                            END
```

```
3307
3308                        ELSE
3309
3310                            BDATA[QTAB_OFFSET]  := TRUE;
3311
3312                    END;        { IF FALSE SDATA[QTAB_OFFSET].DSC$W_LENGTH = 0 }
3313
3314                END;
3315
3316                EDF$K_FILL_LOW,
3317                EDF$K_FILL_HIGH,
3318                EDF$K_DESIRED_FILL :
3319
3320                    IF IDATA[QTAB_OFFSET] < 50 THEN
3321
3322                    BEGIN
3323
3324                        IDATA[QTAB_OFFSET]      := 50;
3325
3326                        IF NOT AUTO_TUNE THEN
3327
3328                        BEGIN
3329
3330                            WRITELN (SHIFT,'Initial Fill of 50% assumed');
3331                            LIB$WAIT (3.0);
3332
3333                        END;
3334
3335                    END;
3336
3337                EDF$K_DESIGN_CYCLE :
3338
3339                BEGIN
3340
3341                    { +
3342                    Make sure he hasn't typed a temporarily disabled option.
3343                    - }
3344                    IF IDATA[EDF$K_SURFACE_OPTION] <> EDF$K_LINE_SURFACE THEN
3345
3346                        IF (
3347                        ((IDATA[EDF$K_SURFACE_OPTION] = EDF$K_FILL_SURFACE)
3348                            AND (INPUT_VALUE = EDF$K_BF))
3349                        OR
3350                        ((IDATA[EDF$K_SURFACE_OPTION] = EDF$K_INIT_SURFACE)
3351                            AND (INPUT_VALUE = EDF$K_IL))
3352                        OR
3353                        ((IDATA[EDF$K_SURFACE_OPTION] = EDF$K_ADDED_SURFACE)
3354                            AND (INPUT_VALUE = EDF$K_AR))
3355                        OR
3356                        ((IDATA[EDF$K_SURFACE_OPTION] = EDF$K_SIZE_SURFACE)
3357                            AND (INPUT_VALUE = EDF$K_RS))
3358                        OR
3359                        ((IDATA[EDF$K_SURFACE_OPTION] = EDF$K_KEY_SURFACE)
3360                            AND (INPUT_VALUE = EDF$K_KL))
3361                        OR
3362                        (
3363                        ( NOT (IDATA[EDF$K_KEY_TYPE] IN [ FDL$C_PAC, FDL$C_STG ]) )
```

```
3364                        AND (INPUT_VALUE = EDF$K_KL)
3365                    )
3366                    ) THEN
3367
3368                        VERIFY_PROCESS      := FALSE;
3369
3370                {+
3371                Make sure that he modifies only reasonable things.
3372                The following options make sense only for the primary key.
3373                -}
3374                IF (
3375                (IDATA[EDF$K_ACTIVE_KEY] <> 0) AND (INPUT_VALUE IN
3376                [ EDF$K_RF, EDF$K_RS, EDF$K_IL, EDF$K_AR, EDF$K_PV, EDF$K_LM ])
3377                ) THEN
3378
3379                    VERIFY_PROCESS          := FALSE;
3380
3381                IF (
3382                (IDATA[EDF$K_PROLOGUE_VERSION] < 3) AND (INPUT_VALUE IN
3383                [ EDF$K_KC, EDF$K_RC, EDF$K_IC ])
3384                ) THEN
3385
3386                    VERIFY_PROCESS          := FALSE;
3387
3388                IF (IDATA[EDF$K_INITIAL_COUNT] < 1) AND
3389                (INPUT_VALUE = EDF$K_LM) THEN
3390
3391                    VERIFY_PROCESS          := FALSE;
3392
3393                    IF (
3394                    (IDATA[EDF$K_INITIAL_COUNT] < 1)
3395                    AND
3396                    (INPUT_VALUE = EDF$K_BF)
3397                    AND
3398                    (NOT AUTO_TUNE)
3399                    ) THEN
3400
3401                    BEGIN
3402
3403                        WRITELN (SHIFT,ANSI_REVERSE,
3404                        ' Fill Factor used is 100% when Initial Load is zero. ',
3405                        ANSI_RESET);
3406                        LIB$WAIT (3.0);
3407
3408                    END;
3409
3410                END;    { EDF$K_DESIGN_CYCLE }
3411
3412                EDF$K_SURFACE_OPTION :
3413
3414                BEGIN
3415
3416                    {+
3417                    Disallow queer options.
3418                    -}
3419                    IF (
3420                        (IDATA[EDF$K_ACTIVE_KEY] <> 0)
```

```
3421                    AND
3422                        (
3423                        (INPUT_VALUE = EDF$K_SIZE_SURFACE)
3424                        OR
3425                        (INPUT_VALUE = EDF$K_INIT_SURFACE)
3426                        OR
3427                        (INPUT_VALUE = EDF$K_ADDED_SURFACE)
3428                        )
3429                    ) THEN
3430
3431                        VERIFY_PROCESS  := FALSE;
3432
3433                END;    { EDF$K_SURFACE_OPTION }
3434
3435                EDF$K_RECORD_FORMAT :
3436
3437                BEGIN
3438
3439                    IDATA[EDF$K_RECORD_FORMAT]  := INPUT_VALUE;
3440
3441                    { +
3442                    Indexed files can have only fixed or variable record format.
3443                    Relative files can't be stream or undefined.
3444                    - }
3445                    IF (
3446                        (
3447                        (ISAM_ORG)
3448                        AND
3449                        (NOT (IDATA[EDF$K_RECORD_FORMAT] IN [ FDL$C_VAR, FDL$C_FIX ]))
3450                        )
3451                    OR
3452                        (
3453                        (IDATA[EDF$K_SCRIPT_OPTION] = EDF$K_REL_DESIGN_FDL)
3454                        AND
3455                        (IDATA[EDF$K_RECORD_FORMAT] IN [ FDL$C_STM, FDL$C_STMCR, FDL$C_STMLF, FDL$C_UDF ])
3456                        )
3457                    ) THEN
3458
3459                        VERIFY_PROCESS  := FALSE;
3460
3461                END;    { EDF$K_RECORD_FORMAT }
3462
3463                EDF$K_TEST_PRIMARY :
3464
3465                BEGIN
3466
3467                    TEST.OBJECT_TYPE            := PRI;
3468                    TEST.PRIMARY                := INPUT_VALUE::PRIMARY_TYPE;
3469                    TEST.PRINUM                 := INPUT_NUMBER;
3470                    DEFAULT_PRINUM              := INPUT_NUMBER;
3471                    ACTIVE_PRIMARY              := TEST.PRIMARY;
3472                    DEFAULT_PRIMARY             := ACTIVE_PRIMARY;
3473                    QTAB[QTAB_OFFSET].DEFAULT   := INPUT_VALUE;
3474
3475                    IF (
3476                    (TEST.PRIMARY = AREA)
3477                    OR
```

```
3478                    (TEST.PRIMARY = KEY)
3479                    ) THEN
3480
3481                    BEGIN
3482
3483                        IF TEST.PRINUM > 254 THEN
3484
3485                            VERIFY_PROCESS      := FALSE;
3486
3487                        SCAN_DEFINITION (TRUE);
3488
3489                        IF (
3490                        (FOUND_AREA)
3491                        AND
3492                        (TEST.PRIMARY = AREA)
3493                        AND
3494                        ((TEST.PRINUM - HIGH_AREA) > 1)
3495                        ) THEN
3496
3497                            VERIFY_PROCESS      := FALSE;
3498
3499                        IF (
3500                        (FOUND_KEY)
3501                        AND
3502                        (TEST.PRIMARY = KEY)
3503                        AND
3504                        ((TEST.PRINUM - HIGH_KEY) > 1)
3505                        ) THEN
3506
3507                            VERIFY_PROCESS      := FALSE;
3508
3509                        IF (
3510                        (NOT FOUND_AREA)
3511                        AND
3512                        (TEST.PRIMARY = AREA)
3513                        AND
3514                        (TEST.PRINUM > 0)
3515                        ) THEN
3516
3517                            VERIFY_PROCESS      := FALSE;
3518
3519                        IF (
3520                        (NOT FOUND_KEY)
3521                        AND
3522                        (TEST.PRIMARY = KEY)
3523                        AND
3524                        (TEST.PRINUM > 0)
3525                        ) THEN
3526
3527                            VERIFY_PROCESS      := FALSE;
3528
3529                    END { IF TRUE (TEST.PRIMARY = AREA) OR (TEST.PRIMARY = KEY) }
3530
3531                    ELSE
3532
3533                        TEST.PRINUM             := 0;
3534
```

```
3535                    { +
3536                    If we're asking for only ones that exist, make sure this one does.
3537                    - }
3538                    IF NOT FULL_CHOICE THEN
3539
3540                    BEGIN
3541
3542                        DEF_CURRENT     := DEF_HEAD;
3543
3544                        REPEAT
3545
3546                            IF NOT CURRENT_EQ_TEST(TEST,FALSE) THEN
3547
3548                                INCR_CURRENT;
3549
3550                        UNTIL (CURRENT_EQ_TEST(TEST,FALSE) OR (DEF_CURRENT^.FORE = NIL));
3551
3552                        IF DEF_CURRENT <> NIL THEN
3553
3554                        BEGIN
3555
3556                            IF NOT CURRENT_EQ_TEST(TEST,FALSE) THEN
3557
3558                                VERIFY_PROCESS  := FALSE;
3559
3560                        END
3561
3562                        ELSE
3563
3564                            VERIFY_PROCESS      := FALSE;
3565
3566                        END;
3567
3568                    END;    { EDF$K_TEST_PRIMARY }
3569
3570                OTHERWISE
3571
3572                    { NULL-STATEMENT } ;
3573
3574                END;        { CASE }
3575
3576            END;    { VERIFY_PROCESS }
```

```
3578            { ++
3579
3580            POST_PROCESS -- Routine to finish up a question.
3581
3582            This function does any calculations needed once a question is anwerred.
3583
3584            CALLING SEQUENCE:
3585
3586            status := POST_PROCESS;
3587
3588            INPUT PARAMETERS:
3589
3590            none
3591
3592            IMPLICIT INPUTS:
3593
3594            none
3595
3596            OUTPUT PARAMETERS:
3597
3598            none
3599
3600            IMPLICIT OUTPUTS:
3601
3602            none
3603
3604            ROUTINES CALLED:
3605
3606            none
3607
3608            ROUTINE VALUE:
3609
3610            TRUE if we should continue, FALSE otherwise
3611
3612            SIGNALS:
3613
3614            none
3615
3616            SIDE EFFECTS:
3617
3618
3619            -- }
```

```
3621            FUNCTION POST_PROCESS : BOOLEAN;
3622
3623            VAR
3624                I   : INTEGER;
3625
3626            BEGIN
3627
3628                { +
3629                Assume success.
3630                - }
3631                POST_PROCESS        := TRUE;
3632
3633                CASE QTAB_OFFSET OF
3634
3635                    { +
3636                    These are boolean_answer questions.
3637                    - }
3638                    EDF$K_CONFIRM,
3639                    EDF$K_KEY_DUPS,
3640                    EDF$K_SEGMENTED,
3641                    EDF$K_BLOCK_SPAN,
3642                    EDF$K_GLOBAL_WANTED,
3643                    EDF$K_ASCENDING_LOAD,
3644                    EDF$K_ASCENDING_ADDED,
3645                    EDF$K_KEY_COMP_WANTED,
3646                    EDF$K_REC_COMP_WANTED,
3647                    EDF$K_IDX_COMP_WANTED :
3648
3649                    BEGIN
3650
3651                        QUERY_FLAG          := (INPUT_VALUE = EDF$K_YES);
3652                        BDATA[QTAB_OFFSET]  := QUERY_FLAG;
3653
3654                    END;    { BOOLEAN_ANSWER }
3655
3656                    { +
3657                    Generalized answer storage for keyword answers.
3658                    - }
3659                    EDF$K_LOAD_METHOD,
3660                    EDF$K_DESIGN_CYCLE,
3661                    EDF$K_SET_FUNCTION,
3662                    EDF$K_GRANULARITY,
3663                    EDF$K_RESPONSES,
3664                    EDF$K_KEY_TYPE,
3665                    EDF$K_CARR_CTRL :
3666
3667                        IDATA[QTAB_OFFSET]          := INPUT_VALUE;
3668
3669                    { +
3670                    Make the new default whatever the user answers.
3671                    - }
3672                    EDF$K_NUMBER_KEYS,
3673                    EDF$K_SURFACE_OPTION,
3674                    EDF$K_BUCKET_WEIGHT :
3675
3676                    BEGIN
3677
```

```
3678                    IDATA[QTAB_OFFSET]          := INPUT_VALUE;
3679                    QTAB[QTAB_OFFSET].DEFAULT   := INPUT_VALUE;
3680
3681                END;
3682
3683                EDF$K_ANALYSIS :
3684
3685                BEGIN
3686
3687                    ANALYSIS_FILENAME_DESC      := NULL_STRING;
3688                    LIB$SCOPY_DXDX (SDATA[EDF$K_ANALYSIS],ANALYSIS_FILENAME_DESC);
3689                    ANALYSIS_SPECIFIED          := TRUE;
3690
3691                END;
3692
3693                EDF$K_SET_OUTPUT :
3694
3695                BEGIN
3696
3697                    OUTPUT_FILENAME_DESC        := NULL_STRING;
3698                    LIB$SCOPY_DXDX (SDATA[EDF$K_OUTPUT],OUTPUT_FILENAME_DESC);
3699
3700                END;
3701
3702                EDF$K_PROMPTING :
3703
3704                BEGIN
3705
3706                    IDATA[QTAB_OFFSET]          := INPUT_VALUE;
3707                    FULL_PROMPT                 := (INPUT_VALUE = EDF$K_FULL);
3708
3709                END;
3710
3711                EDF$K_SCRIPT_OPTION :
3712
3713                BEGIN
3714
3715                    IDATA[QTAB_OFFSET]          := INPUT_VALUE;
3716
3717                    ISAM_ORG    := (INPUT_VALUE IN [ EDF$K_IDX_DESIGN_FDL,
3718                                    EDF$K_ADD_KEY_FDL, EDF$K_DELETE_KEY_FDL,
3719                                    EDF$K_REDESIGN_FDL, EDF$R_OPTIMIZE_FDL ]);
3720
3721                END;    { SCRIPT_OPTION }
3722
3723                { +
3724                These are the real_answer questions.
3725                - }
3726                EDF$K_DATA_KEY_COMP,
3727                EDF$K_DATA_RECORD_COMP,
3728                EDF$K_INDEX_RECORD_COMP :
3729
3730                BEGIN
3731
3732                    { +
3733                    Make sure we aren't fooled.
3734                    - }
```

```
3735                    IF ABS (IDATA[QTAB_OFFSET]) > 99 THEN
3736
3737                        IDATA[QTAB_OFFSET]      := 0;
3738
3739                    RDATA[QTAB_OFFSET]  := IDATA[QTAB_OFFSET];
3740                    RDATA[QTAB_OFFSET]  := RDATA[QTAB_OFFSET] / 100.0;
3741
3742                END;    { EDF$K_DATA_KEY_COMP }
3743
3744                EDF$K_RETURN :
3745
3746                BEGIN
3747
3748                    IF NOT AUTO_TUNE THEN
3749
3750                    BEGIN
3751
3752                        { +
3753                        Now that he's answered, clear his screen.
3754                        - }
3755                        IF REGIS THEN
3756
3757                            WRITELN (''(27)'Pp;S(E);'(27)'\');
3758
3759                        LIB$ERASE_PAGE (LINE_ONE,COL_ONE);
3760
3761                    END;        { IF NOT AUTO_TUNE }
3762
3763                END;    { EDF$K_RETURN }
3764
3765                EDF$K_GLOBAL_COUNT :
3766
3767                BEGIN
3768
3769                    { +
3770                    GLOBAL_SET is true if GLOBAL_COUNT is set from the
3771                    definition linked list.
3772                    - }
3773                    IF NOT GLOBAL_SET THEN
3774
3775                    BEGIN
3776
3777                        { +
3778                        See how many global buffers would map the entire key 0
3779                        index, plus the roots of all the alternate
3780                        keys, plus 5 data buckets. (why 5? it sounds good...)
3781                        If a level has more than 512 buckets, only 512 are counted.
3782                        (let's not get ridiculous here)
3783                        - }
3784                        PRIMARY_INDEX_BUCKETS   := 0;
3785
3786                        FOR I := 1 TO 31 DO
3787
3788                        BEGIN
3789
3790                            IF INIT_PRIMARY_BUCKETS [I] > 512 THEN
3791
```

```
3792                        INIT_PRIMARY_BUCKETS [I]        := 512;
3793
3794                    IF ADDED_PRIMARY_BUCKETS [I] > 512 THEN
3795
3796                        ADDED_PRIMARY_BUCKETS [I]       := 512;
3797
3798                    PRIMARY_INDEX_BUCKETS       := PRIMARY_INDEX_BUCKETS
3799                                                    + INIT_PRIMARY_BUCKETS [I]
3800                                                    + ADDED_PRIMARY_BUCKETS [I];
3801
3802                    END;
3803
3804                    { +
3805                    4 is added instead of 5 so we don't have to use
3806                    (idata[edf$K_number_keys]-1) for the number of alternate keys.
3807                    - }
3808                    IDATA[EDF$K_GLOBAL_COUNT]       := PRIMARY_INDEX_BUCKETS +
3809                                                    IDATA[EDF$K_NUMBER_KEYS] + 4;
3810
3811                END;
3812
3813                { +
3814                Up to an RMS maximum.
3815                - }
3816                IF IDATA[EDF$K_GLOBAL_COUNT] > EDF$C_MAX_GBL_BUFS THEN
3817
3818                    IDATA[EDF$K_GLOBAL_COUNT]       := EDF$C_MAX_GBL_BUFS;
3819
3820            END;    { EDF$K_GLOBAL_COUNT }
3821
3822            EDF$K_NUMBER_RECORDS :
3823
3824                IDATA[EDF$K_INITIAL_COUNT]  := IDATA[QTAB_OFFSET];
3825
3826            EDF$K_KEY_POSITION :
3827
3828                SEGMENT_POSITION[SEGMENT_NUMBER] := IDATA[QTAB_OFFSET];
3829
3830            EDF$K_KEY_SIZE :
3831
3832            BEGIN
3833
3834                SEGMENT_WANTED[SEGMENT_NUMBER] := (IDATA[EDF$K_KEY_SIZE] > 0);
3835                SEGMENT_LENGTH[SEGMENT_NUMBER] := IDATA[QTAB_OFFSET];
3836
3837            END;
3838
3839            EDF$K_CONTROL_SIZE :
3840
3841                CUR_MAX_REC         := CUR_MAX_REC - IDATA[QTAB_OFFSET];
3842
3843            EDF$K_NUMBER_DUPS :
3844
3845                IF IDATA[QTAB_OFFSET] < 0 THEN
3846
3847                    IDATA[QTAB_OFFSET]      := 0;
3848
```

```
3849                    EDF$K_PROLOGUE_VERSION :
3850
3851                    BEGIN
3852
3853                        IF IDATA[EDF$K_PROLOGUE_VERSION] < 3 THEN
3854
3855                        BEGIN
3856
3857                            RDATA[EDF$K_DATA_RECORD_COMP]   := 0.0;
3858                            RDATA[EDF$K_DATA_KEY_COMP]      := 0.0;
3859                            RDATA[EDF$K_INDEX_RECORD_COMP]  := 0.0;
3860
3861                        END;
3862
3863                    END;    { EDF$K_PROLOGUE_VERSION }
3864
3865                    EDF$K_KEY_LOW,
3866                    EDF$K_ADDED_COUNT_LOW,
3867                    EDF$K_INITIAL_COUNT_LOW,
3868                    EDF$K_SIZE_LOW,
3869                    EDF$K_FILL_LOW :
3870
3871                    BEGIN
3872
3873                        IDATA[EDF$K_Y_LOW]                  := IDATA[QTAB_OFFSET];
3874                        QTAB[QTAB_OFFSET+1].LOW_BOUND       := IDATA[QTAB_OFFSET];
3875
3876                    END;
3877
3878                    EDF$K_KEY_HIGH,
3879                    EDF$K_FILL_HIGH,
3880                    EDF$K_SIZE_HIGH,
3881                    EDF$K_ADDED_COUNT_HIGH,
3882                    EDF$K_INITIAL_COUNT_HIGH :
3883
3884                    BEGIN
3885
3886                        IDATA[EDF$K_Y_HIGH]         := IDATA[QTAB_OFFSET];
3887
3888                        CASE QTAB_OFFSET OF
3889
3890                            EDF$K_FILL_HIGH :       SPREAD_LOW_HIGH (50,100);
3891
3892                            EDF$K_SIZE_HIGH :       SPREAD_LOW_HIGH (1,CUR_MAX_REC);
3893
3894                            EDF$K_KEY_HIGH,
3895                            EDF$K_ADDED_COUNT_HIGH,
3896                            EDF$K_INITIAL_COUNT_HIGH :      SPREAD_LOW_HIGH (1,MAXINT-1);
3897
3898                        OTHERWISE
3899
3900                            { NULL-STATEMENT } ;
3901
3902                        END;        { CASE }
3903
3904                    END;
3905
```

```
3906                    EDF$K_DESIRED_FILL :
3907
3908                        IDATA[EDF$K_FDL_FILL]       := IDATA[EDF$K_DESIRED_FILL];
3909
3910                    EDF$K_CURRENT_FUNCTION :
3911
3912                    BEGIN
3913
3914                        IDATA[QTAB_OFFSET]          := INPUT_VALUE;
3915
3916                        { +
3917                        Reset the script pointer (only auto-invoke on 1st entry).
3918                        - }
3919                        IDATA[EDF$K_SCRIPT_OPTION]  := EDF$K_ZERO_SCRIPT;
3920
3921                        { +
3922                        Reset the ^Z flag.
3923                        - }
3924                        MAIN_LEVEL                  := FALSE;
3925
3926                    END;    { EDF$K_CURRENT_FUNCTION }
3927
3928                    EDF$K_RECORD_FORMAT :
3929
3930                    BEGIN
3931
3932                        { +
3933                        The IDATA[EDF$K_RECORD_FORMAT] variable was set in VERIFY_PROCESS.
3934                        - }
3935
3936                        { +
3937                        Set a convenience boolean.
3938                        - }
3939                        VARIABLE_RECORDS    := (
3940                                               (IDATA[EDF$K_RECORD_FORMAT] <> FDL$C_FIX)
3941                                               AND
3942                                               (IDATA[EDF$K_RECORD_FORMAT] <> FDL$C_UDF)
3943                                               );
3944
3945                    END;    { EDF$K_RECORD_FORMAT }
3946
3947                    EDF$K_TEST_PRIMARY :
3948
3949                    BEGIN
3950
3951                        IF ACTIVE_PRIMARY = AREA THEN
3952
3953                            ACTIVE_AREA             := INPUT_NUMBER
3954
3955                        ELSE IF ACTIVE_PRIMARY = KEY THEN
3956
3957                            IDATA[EDF$K_ACTIVE_KEY] := INPUT_NUMBER;
3958
3959                    END;    { EDF$K_TEST_PRIMARY }
3960
3961                OTHERWISE
3962
```

```
3963                    { NULL-STATEMENT } ;
3964
3965                END;        { CASE }
3966
3967            END;    { POST_PROCESS }
```

```
3969          { ++
3970
3971          QUERY -- Routine to control the asking of questions.
3972
3973          This function processes the QTAB table, and interacts with the user.
3974
3975          CALLING SEQUENCE:
3976
3977          status := QUERY (QTAB-OFFSET-VALUE);
3978
3979          INPUT PARAMETERS:
3980
3981          none
3982
3983          IMPLICIT INPUTS:
3984
3985          none
3986
3987          OUTPUT PARAMETERS:
3988
3989          none
3990
3991          IMPLICIT OUTPUTS:
3992
3993          SYS$OUTPUT:
3994
3995          ROUTINES CALLED:
3996
3997          WRITE_QUESTION
3998          WRITE_HELP
3999          PRE_PROCESS
4000          VERIFY_PROCESS
4001          POST_PROCESS
4002
4003          ROUTINE VALUE:
4004
4005          TRUE if answer was yes, FALSE otherwise
4006
4007          SIGNALS:
4008
4009          none
4010
4011          SIDE EFFECTS:
4012
4013
4014          -- }
```

```
4016        [GLOBAL] FUNCTION QUERY (OFFSET : INTEGER) : BOOLEAN;
4017
4018            PROCEDURE THE_QUESTION;
4019
4020            BEGIN
4021
4022                { +
4023                Special for top level query.
4024                - }
4025                IF QTAB_OFFSET = EDF$K_CURRENT_FUNCTION THEN
4026
4027                BEGIN
4028
4029                    MAIN_LEVEL          := TRUE;
4030                    MAIN_CTRLZ          := FALSE;
4031                    CONTROL_ZEE_TYPED   := FALSE;
4032
4033                END;    { IF TRUE QTAB_OFFSET = EDF$K_CURRENT_FUNCTION }
4034
4035                { +
4036                Setup to catch bad user input.
4037                - }
4038                SYS$INPUT_ERROR := FALSE;
4039                ESTABLISH (SYS$INPUT_COND_HANDLER);
4040
4041                IF NOT AUTO_TUNE THEN
4042
4043                BEGIN
4044
4045                    IF TEMP_FULL_PROMPT THEN
4046
4047                        WRITE_HELP;
4048
4049                    WRITE_QUESTION;
4050
4051                END;
4052
4053                CASE QTAB[QTAB_OFFSET].ANSWER_CLASS OF
4054
4055                    STRING_ANSWER :
4056
4057                    BEGIN
4058
4059                        SDATA[QTAB_OFFSET]      := NULL_STRING;
4060
4061                        IF (
4062                        (
4063                        (TAKE_DEFAULTS)
4064                        AND
4065                        (IDATA[EDF$K_RESPONSES] = EDF$K_AUTO)
4066                        )
4067                        OR
4068                        (AUTO_TUNE)
4069                        ) THEN
4070
4071                        BEGIN
4072
```

```
4073                        IF NOT AUTO_TUNE THEN
4074
4075                            LIB$WAIT (0.7);
4076
4077                        END
4078
4079                        ELSE
4080
4081                        BEGIN
4082
4083                            IF EOF (INPUT) THEN
4084
4085                            BEGIN
4086
4087                                RESET (INPUT);
4088                                LIB$SIGNAL (EDF$_CTRLZ,0,0,0);
4089
4090                            END;
4091
4092                            READLN (TEMP_STRING255);
4093                            STR$TRIM (SDATA[QTAB_OFFSET],TEMP_STRING255);
4094                            LIB$SCOPY_DXDX (SDATA[QTAB_OFFSET],INPUT_DESC);
4095                            PARAM_BLOCK.TPA$L_TOKENPTR  := INPUT_DESC.DSC$A_POINTER::UNSIGNED;
4096                            PARAM_BLOCK.TPA$L_TOKENCNT  := INPUT_DESC.DSC$W_LENGTH;
4097
4098                        END;
4099
4100                        IF NOT AUTO_TUNE THEN
4101
4102                            WRITELN (CRLF);
4103
4104                        { +
4105                        If we're journaling our input, save a copy of it to the
4106                        journal file.
4107                        - }
4108                        IF JOURNAL_ENABLED THEN
4109
4110                            IF SDATA[QTAB_OFFSET].DSC$W_LENGTH > 0 THEN
4111
4112                                WRITELN (
4113                                    JOURNAL_FILE,
4114                                    SDATA[QTAB_OFFSET].DSC$A_POINTER^:
4115                                        SDATA[QTAB_OFFSET].DSC$W_LENGTH
4116                                    )
4117
4118                            ELSE
4119
4120                                WRITELN (JOURNAL_FILE);
4121
4122                    END;        { STRING_ANSWER }
4123
4124                    REAL_ANSWER,        { Actually, real_answer = integer percentage }
4125                    INTEGER_ANSWER :
4126
4127                    BEGIN
4128
4129                        NUMBER_INPUT (
```

```
4130                            IDATA[QTAB_OFFSET],
4131                            QTAB[QTAB_OFFSET].DEFAULT_OK,
4132                            QTAB[QTAB_OFFSET].DEFAULT);
4133
4134                    { +
4135                    Max record size of 0 is one case where it's OK to
4136                    specify an answer that's out of the low-high range.
4137                    - }
4138                    IF (
4139                    (
4140                        (QTAB_OFFSET = EDF$K_MAX_RECORD_SIZE)
4141                        AND
4142                        (IDATA[EDF$K_SCRIPT_OPTION] <> EDF$K_REL_DESIGN_FDL)
4143                        AND
4144                        (IDATA[QTAB_OFFSET] = 0)
4145                    )
4146                    OR
4147                    (
4148                        (BDATA[EDF$K_SEGMENTED])
4149                        AND
4150                        (SEGMENT_NUMBER <> 0)
4151                        AND
4152                        (QTAB_OFFSET = EDF$K_KEY_SIZE)
4153                        AND
4154                        (IDATA[QTAB_OFFSET] = 0)
4155                    )
4156                    ) THEN
4157
4158                    BEGIN
4159
4160                        { NULL-STATEMENT } ;
4161
4162                    END
4163
4164                    ELSE
4165
4166                    BEGIN
4167
4168                        IF (
4169                        (IDATA[QTAB_OFFSET] < QTAB[QTAB_OFFSET].LOW_BOUND)
4170                        OR
4171                        (IDATA[QTAB_OFFSET] > QTAB[QTAB_OFFSET].HIGH_BOUND)
4172                        ) THEN
4173
4174                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
4175
4176                    END;
4177
4178                END;        { REAL_ANSWER,INTEGER_ANSWER }
4179
4180                BOOLEAN_ANSWER,
4181                KEYWORD_ANSWER :
4182
4183                BEGIN
4184
4185                    PARSE_INPUT (
4186                            QTAB[QTAB_OFFSET].KEY_TABLE,
```

```
4187                        QTAB[QTAB_OFFSET].STATE_TABLE,
4188                        QTAB[QTAB_OFFSET].DEFAULT_OK,
4189                        QTAB[QTAB_OFFSET].DEFAULT);
4190
4191                END;        { BOOLEAN_ANSWER, KEYWORD_ANSWER }
4192
4193                NO_ANSWER :
4194
4195                BEGIN
4196
4197                    { *
4198                    When the user just types <CR>, then accept anything.
4199                    - }
4200                    IF AUTO_TUNE THEN
4201
4202                    BEGIN
4203
4204                        { NULL-STATEMENT } ;
4205
4206                    END
4207
4208                    ELSE
4209
4210                    BEGIN
4211
4212                        IF EOF (INPUT) THEN
4213
4214                        BEGIN
4215
4216                            RESET (INPUT);
4217                            LIB$SIGNAL (EDF$_CTRLZ,0,0,0);
4218
4219                        END;
4220
4221                        READLN;
4222
4223                        IF JOURNAL_ENABLED THEN
4224
4225                            WRITELN (JOURNAL_FILE);
4226
4227                    END;
4228
4229                END;        { NO_ANSWER }
4230
4231                OBJECT_ANSWER :
4232
4233                    { T.B.S. } ;
4234
4235            OTHERWISE
4236
4237                { NULL-STATEMENT } ;
4238
4239            END;    { CASE }
4240
4241            { *
4242            Do some initial checking of the answer.
4243            - }
```

```
4244                IF NOT VERIFY_PROCESS THEN
4245
4246                    LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
4247
4248                { +
4249                Make sure this is true for only one cycle.
4250                - }
4251                TEMP_FULL_PROMPT        := FALSE;
4252
4253            END;        { THE_QUESTION }
```

```
4255            BEGIN
4256
4257                { +
4258                Make which question we're on widely known.
4259                - }
4260                QTAB_OFFSET         := OFFSET;
4261
4262                IF PRE_PROCESS THEN
4263
4264                BEGIN
4265
4266                    { +
4267                    Keep at it until the user gets it right.
4268                    - }
4269                    REPEAT
4270
4271                        THE_QUESTION;
4272
4273                    UNTIL NOT SYS$INPUT_ERROR;
4274
4275                    STR$FREE1_DX (INPUT_DESC);
4276
4277                END;        { IF TRUE PRE_PROCESS }
4278
4279                { +
4280                If this question has a valid answer, flag it so.
4281                - }
4282                IF POST_PROCESS THEN
4283
4284                    VDATA[QTAB_OFFSET]      := TRUE;
4285
4286                { +
4287                Set the output function value.
4288                - }
4289                QUERY       := QUERY_FLAG;
4290
4291            END;    { QUERY }
```

```
4293            { ++
4294
4295            ASK_KEY_DUPS -- Query the user.
4296
4297            This routine asks the user if he wants duplicates on his key.
4298            If he does, then it asks him how many.
4299
4300            CALLING SEQUENCE:
4301
4302            ASK_KEY_DUPS;
4303
4304            INPUT PARAMETERS:
4305
4306            none
4307
4308            IMPLICIT INPUTS:
4309
4310            SYS$INPUT_ERROR
4311            SYS$INPUT:
4312
4313            OUTPUT PARAMETERS:
4314
4315            none
4316
4317            IMPLICIT OUTPUTS:
4318
4319            BDATA[EDF$K_NUMBER_DUPS]
4320            BDATA[EDF$K_KEY_DUPS]
4321
4322            ROUTINES CALLED:
4323
4324            none
4325
4326            ROUTINE VALUE:
4327
4328            none
4329
4330            SIGNALS:
4331
4332            none
4333
4334            SIDE EFFECTS:
4335
4336            none
4337
4338            -- }
```

```
4340            PROCEDURE ASK_KEY_DUPS;
4341
4342            BEGIN
4343
4344                IF QUERY (EDF$K_KEY_DUPS) THEN
4345
4346                    QUERY (EDF$K_NUMBER_DUPS)
4347
4348                ELSE
4349
4350                    IDATA[EDF$K_NUMBER_DUPS]        := 0;
4351
4352            END;    { ASK_KEY_DUPS }
```

```
4354          { ++
4355
4356          ASK_GLOBAL_WANTED -- Query the user.
4357
4358          This routine asks the user if he wants Global Buffers, and if he does, then
4359          it asks him how many.
4360
4361          CALLING SEQUENCE:
4362
4363          ASK_GLOBAL_WANTED;
4364
4365          INPUT PARAMETERS:
4366
4367          none
4368
4369          IMPLICIT INPUTS:
4370
4371          SYS$INPUT_ERROR
4372
4373          OUTPUT PARAMETERS:
4374
4375          none
4376
4377          IMPLICIT OUTPUTS:
4378
4379          IDATA[EDF$K_GLOBAL_COUNT]
4380          BDATA[EDF$K_GLOBAL_WANTED]
4381
4382          ROUTINES CALLED:
4383
4384          QUERY (EDF$K_GLOBAL_WANTED)
4385          QUERY (EDF$K_GLOBAL_COUNT)
4386
4387          ROUTINE VALUE:
4388
4389          none
4390
4391          SIGNALS:
4392
4393          none
4394
4395          SIDE EFFECTS:
4396
4397          none
4398
4399          -- }
```

```
4401            PROCEDURE ASK_GLOBAL_WANTED;
4402
4403            BEGIN
4404
4405                { +
4406                If we want Global Buffers, see how many.
4407                - }
4408                IF QUERY (EDF$K_GLOBAL_WANTED) THEN
4409
4410                    QUERY (EDF$K_GLOBAL_COUNT)
4411
4412                ELSE
4413
4414                    IDATA[EDF$K_GLOBAL_COUNT]       := 0;
4415
4416            END;    { ASK_GLOBAL_WANTED }
```

```
4418          { ++
4419
4420          ASK_KEY_COMP -- Query the user.
4421
4422          This routine asks the user if he wants key compression and if he does, then
4423          it finds out what the compression was.
4424
4425          CALLING SEQUENCE:
4426
4427          ASK_KEY_COMP;
4428
4429          INPUT PARAMETERS:
4430
4431          none
4432
4433          IMPLICIT INPUTS:
4434
4435          SYS$INPUT_ERROR
4436
4437          OUTPUT PARAMETERS:
4438
4439          none
4440
4441          IMPLICIT OUTPUTS:
4442
4443          RDATA[EDF$K_DATA_KEY_COMP]
4444          BDATA[EDF$K_KEY_COMP_WANTED]
4445
4446          ROUTINES CALLED:
4447
4448          QUERY (EDF$K_KEY_COMP_WANTED)
4449          QUERY (EDF$K_DATA_KEY_COMP)
4450
4451          ROUTINE VALUE:
4452
4453          none
4454
4455          SIGNALS:
4456
4457          none
4458
4459          SIDE EFFECTS:
4460
4461          none
4462
4463          -- }
```

```
4465            PROCEDURE ASK_KEY_COMP;
4466
4467            BEGIN
4468
4469                { +
4470                If we want compression. See what it is.
4471                - }
4472                IF QUERY (EDF$K_KEY_COMP_WANTED) THEN
4473
4474                    QUERY (EDF$K_DATA_KEY_COMP)
4475
4476                ELSE
4477
4478                    RDATA[EDF$K_DATA_KEY_COMP]      := 0.0;
4479
4480            END;    { ASK_KEY_COMP }
```

```
4482            { ++
4483
4484            ASK_REC_COMP -- Query the user.
4485
4486            This routine asks the user if he wants record compression, and if he does, then
4487            it finds out how much there is.
4488
4489            CALLING SEQUENCE:
4490
4491            ASK_REC_COMP;
4492
4493            INPUT PARAMETERS:
4494
4495            none
4496
4497            IMPLICIT INPUTS:
4498
4499            SYS$INPUT_ERROR
4500
4501            OUTPUT PARAMETERS:
4502
4503            none
4504
4505            IMPLICIT OUTPUTS:
4506
4507            RDATA[EDF$K_DATA_RECORD_COMP]
4508            BDATA[EDF$K_REC_COMP_WANTED]
4509
4510            ROUTINES CALLED:
4511
4512            QUERY (EDF$K_REC_COMP_WANTED)
4513            QUERY (EDF$K_DATA_RECORD_COMP)
4514
4515            ROUTINE VALUE:
4516
4517            none
4518
4519            SIGNALS:
4520
4521            none
4522
4523            SIDE EFFECTS:
4524
4525            none
4526
4527            -- }
```

```
4529            PROCEDURE ASK_REC_COMP;
4530
4531            BEGIN
4532
4533                { +
4534                If we want compression. See what it is.
4535                - }
4536                IF QUERY (EDF$K_REC_COMP_WANTED) THEN
4537
4538                    QUERY (EDF$K_DATA_RECORD_COMP)
4539
4540                ELSE
4541
4542                    RDATA[EDF$K_DATA_RECORD_COMP]   := 0.0;
4543
4544            END;    { ASK_REC_COMP }
```

```
4546            { ++
4547
4548            ASK_IDX_COMP -- Query the user.
4549
4550            This routine asks the user if he wants index compression and if he does, then
4551            it finds out how much there is.
4552
4553            CALLING SEQUENCE:
4554
4555            ASK_IDX_COMP;
4556
4557            INPUT PARAMETERS:
4558
4559            none
4560
4561            IMPLICIT INPUTS:
4562
4563            SYS$INPUT_ERROR
4564
4565            OUTPUT PARAMETERS:
4566
4567            none
4568
4569            IMPLICIT OUTPUTS:
4570
4571            RDATA[EDF$K_INDEX_RECORD_COMP]
4572            BDATA[EDF$K_IDX_COMP_WANTED]
4573
4574            ROUTINES CALLED:
4575
4576            QUERY (EDF$K_IDX_COMP_WANTED)
4577            QUERY (EDF$K_INDEX_RECORD_COMP)
4578
4579            ROUTINE VALUE:
4580
4581            none
4582
4583            SIGNALS:
4584
4585            none
4586
4587            SIDE EFFECTS:
4588
4589            none
4590
4591            -- }
```

```
4593            PROCEDURE ASK_IDX_COMP;
4594
4595            BEGIN
4596
4597                {+
4598                If we want compression. See what it is.
4599                -}
4600                IF QUERY (EDF$K_IDX_COMP_WANTED) THEN
4601
4602                    QUERY (EDF$K_INDEX_RECORD_COMP)
4603
4604                ELSE
4605
4606                    RDATA[EDF$K_INDEX_RECORD_COMP]  := 0.0;
4607
4608            END;    { ASK_IDX_COMP }
```

```
4610          { ++
4611
4612          ASK_MEAN_RECORD_SIZE -- Query the user.
4613
4614          This routine asks about the user's record size. (plus max_rec, and control_size)
4615
4616          CALLING SEQUENCE:
4617
4618          ASK_MEAN_RECORD_SIZE;
4619
4620          INPUT PARAMETERS:
4621
4622          none
4623
4624          IMPLICIT INPUTS:
4625
4626          SYS$INPUT_ERROR
4627
4628          OUTPUT PARAMETERS:
4629
4630          none
4631
4632          IMPLICIT OUTPUTS:
4633
4634          IDATA[EDF$K_MEAN_RECORD_SIZE]
4635          SYS$INPUT_ERROR
4636
4637          ROUTINES CALLED:
4638
4639          ESTABLISH
4640
4641          ROUTINE VALUE:
4642
4643          none
4644
4645          SIGNALS:
4646
4647          none
4648
4649          SIDE EFFECTS:
4650
4651          none
4652
4653          -- }
```

```
4655        PROCEDURE ASK_MEAN_RECORD_SIZE;
4656
4657        BEGIN
4658
4659            { +
4660            This question shouldn't be asked for alternate keys. Unless redesigning,
4661            and we don't already have a value for it.
4662            - }
4663            IF (
4664            (NOT ISAM_ORG)
4665            OR
4666            (ISAM_ORG AND (IDATA[EDF$K_ACTIVE_KEY] = 0))
4667            OR
4668            (NOT VDATA[EDF$K_MEAN_RECORD_SIZE])
4669            ) THEN
4670
4671            BEGIN
4672
4673                { +
4674                Ask the question we're here for.
4675                - }
4676                QUERY (EDF$K_MEAN_RECORD_SIZE);
4677
4678                { +
4679                Get (or set) the other record size parameter.
4680
4681                THE CONTROL_SIZE QUESTION MUST COME BEFORE THE MAX_RECORD_SIZE
4682                QUESTION!
4683
4684                - }
4685                IF IDATA[EDF$K_RECORD_FORMAT] = FDL$C_VFC THEN
4686
4687                    QUERY (EDF$K_CONTROL_SIZE);
4688
4689                IF VARIABLE_RECORDS THEN
4690
4691                    QUERY (EDF$K_MAX_RECORD_SIZE)
4692
4693                ELSE
4694
4695                    IDATA[EDF$K_MAX_RECORD_SIZE] := IDATA[EDF$K_MEAN_RECORD_SIZE];
4696
4697            END;
4698
4699        END;    { ASK_MEAN_RECORD_SIZE }
```

```
4701            { ++
4702
4703            ASK_KEY_SIZE -- Query the user.
4704
4705            This routine asks about the user's key size.
4706
4707            CALLING SEQUENCE:
4708
4709            ASK_KEY_SIZE;
4710
4711            INPUT PARAMETERS:
4712
4713            none
4714
4715            IMPLICIT INPUTS:
4716
4717            SYS$INPUT_ERROR
4718
4719            OUTPUT PARAMETERS:
4720
4721            none
4722
4723            IMPLICIT OUTPUTS:
4724
4725            IDATA[EDF$K_KEY_SIZE]
4726            SEGMENT_LENGTH[0..7]
4727            SYS$INPUT_ERROR
4728
4729            ROUTINES CALLED:
4730
4731            ESTABLISH
4732
4733            ROUTINE VALUE:
4734
4735            none
4736
4737            SIGNALS:
4738
4739            none
4740
4741            SIDE EFFECTS:
4742
4743            none
4744
4745            -- }
```

```
4747        PROCEDURE ASK_KEY_SIZE;
4748
4749        VAR
4750            I   : INTEGER;
4751
4752        BEGIN
4753
4754            IF BDATA[EDF$K_SEGMENTED] THEN
4755
4756            BEGIN
4757
4758                SEGMENT_NUMBER  := 0;
4759
4760                REPEAT
4761
4762                    QUERY (EDF$K_KEY_SIZE);
4763
4764                    SEGMENT_NUMBER      := SEGMENT_NUMBER + 1;
4765
4766                UNTIL (IDATA[EDF$K_KEY_SIZE] = 0) OR (SEGMENT_NUMBER > 7);
4767
4768                IF IDATA[EDF$K_KEY_SIZE] = 0 THEN
4769
4770                BEGIN
4771
4772                    FOR I := SEGMENT_NUMBER TO 7 DO
4773
4774                        SEGMENT_WANTED[I]       := FALSE;
4775
4776                    END;
4777
4778                    IDATA[EDF$K_KEY_SIZE]   := SEGMENT_LENGTH[0];
4779
4780                END
4781
4782                ELSE
4783
4784                    QUERY (EDF$K_KEY_SIZE);
4785
4786            END;    { ASK_KEY_SIZE }
```

```
4788            { ++
4789
4790            ASK_KEY_POSITION -- Query the user.
4791
4792            This routine asks about the user's key position.
4793
4794            CALLING SEQUENCE:
4795
4796            ASK_KEY_POSITION;
4797
4798            INPUT PARAMETERS:
4799
4800            none
4801
4802            IMPLICIT INPUTS:
4803
4804            SYS$INPUT_ERROR
4805
4806            OUTPUT PARAMETERS:
4807
4808            none
4809
4810            IMPLICIT OUTPUTS:
4811
4812            IDATA[EDF$K_KEY_POSITION]
4813            SEGMENT_POSITION[0..7]
4814            SYS$INPUT_ERROR
4815
4816            ROUTINES CALLED:
4817
4818            ESTABLISH
4819
4820            ROUTINE VALUE:
4821
4822            none
4823
4824            SIGNALS:
4825
4826            none
4827
4828            SIDE EFFECTS
4829
4830            none
4831
4832            -- }
```

```
4834            PROCEDURE ASK_KEY_POSITION;
4835
4836            BEGIN
4837
4838                IF BDATA[EDF$K_SEGMENTED] THEN
4839
4840                BEGIN
4841
4842                    FOR SEGMENT_NUMBER := 0 TO 7 DO
4843
4844                    BEGIN
4845
4846                        IF SEGMENT_WANTED[SEGMENT_NUMBER] THEN
4847
4848                            QUERY (EDF$K_KEY_POSITION);
4849
4850                    END;
4851
4852                    IDATA[EDF$K_KEY_POSITION]       := SEGMENT_POSITION[0];
4853
4854                END
4855
4856                ELSE
4857
4858                    QUERY (EDF$K_KEY_POSITION);
4859
4860            END;    { ASK_KEY_POSITION }
```

```
4862            { ++
4863
4864            ASK_TEST_SECONDARY -- Get the user's choice of secondary.
4865
4866            This routine queries the user.
4867
4868            CALLING SEQUENCE:
4869
4870            ASK_TEST_SECONDARY
4871
4872            INPUT PARAMETERS:
4873
4874            none
4875
4876            IMPLICIT INPUTS:
4877
4878            CRLF
4879            TAB
4880            SYS$INPUT_ERROR
4881            SYS$INPUT:
4882
4883            OUTPUT PARAMETERS:
4884
4885            none
4886
4887            IMPLICIT OUTPUTS:
4888
4889            SYS$OUTPUT:
4890
4891            ROUTINES CALLED:
4892
4893            ESTABLISH
4894
4895            ROUTINE VALUE:
4896
4897            none
4898
4899            SIGNALS:
4900
4901            none
4902
4903            SIDE EFFECTS:
4904
4905            none
4906
4907            -- }
```

```
4909        PROCEDURE ASK_TEST_SECONDARY;
4910
4911            PROCEDURE THE_QUESTION;
4912
4913            BEGIN
4914
4915                { +
4916                Set up to catch bad user input.
4917                - }
4918                EDF$GL_SECNUM    := 0;
4919                SYS$INPUT_ERROR := FALSE;
4920                ESTABLISH (SYS$INPUT_COND_HANDLER);
4921
4922                CASE ACTIVE_PRIMARY OF
4923
4924                    IDENT :
4925
4926                        INPUT_VALUE      := 0;    { DUMMY_SECONDARY$ }
4927
4928                    TITLE :
4929
4930                        INPUT_VALUE      := 0;    { DUMMY_SECONDARY$ }
```

```
4932            ACCESS :
4933
4934            BEGIN
4935
4936                    IF FULL_CHOICE THEN
4937
4938                        BEGIN
4939
4940                            CLEAR (IF_FULL_PROMPT);
4941
4942                            IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
4943
4944                            BEGIN
4945
4946                                WRITELN (SHIFT,'                ',ANSI_REVERSE,
4947                                ' Legal ACCESS',SEC_ATTR,
4948                                ANSI_RESET,CRLF,
4949                                CRLF_SHIFT,
4950                                'BLOCK_IO               yes/no',
4951                                CRLF_SHIFT,
4952                                'DELETE                 yes/no',
4953                                CRLF_SHIFT
4954                                'GET                    yes/no',
4955                                CRLF_SHIFT,
4956                                'PUT                    yes/no',
4957                                CRLF_SHIFT,
4958                                'RECORD_IO              yes/no',
4959                                CRLF_SHIFT,
4960                                'TRUNCATE               yes/no',
4961                                CRLF_SHIFT,
4962                                'UPDATE                 yes/no',
4963                                CRLF);
4964
4965                            END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
4966
4967                            ELSE
4968
4969                                WRITELN (SHIFT,QUES_HINT);
4970
4971                        END
4972
4973                ELSE
4974
4975                        BEGIN
4976
4977                            CLEAR (IF_FULL_PROMPT);
4978
4979                            IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
4980
4981                            BEGIN
4982
4983                                WRITELN (SHIFT,'                ',ANSI_REVERSE,
4984                                ' Current ACCESS',SEC_ATTR,
4985                                ANSI_RESET,CRLF);
4986
4987                                { +
4988                                Setup to display definition on the terminal.
```

```
4989                                    - }
4990                                    OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
4991                                            RECORD_LENGTH := 252);
4992                                    REWRITE (FDL_DEST);
4993
4994                                    SHOW_PRIMARY_SECTION (TEST);
4995
4996                                    CLOSE (FDL_DEST);
4997
4998                                END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
4999
5000                                ELSE
5001
5002                                    WRITELN (SHIFT,QUES_HINT);
5003
5004                            END;    { EXTANT_ONLY }
5005
5006                        WRITE (SHIFT,'Enter ACCESS Attribute            (Keyword)',
5007                        ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5008                        PARSE_INPUT (
5009                                    IADDRESS (EDF$AB_ACCESS_TABLE_KEY),
5010                                    IADDRESS (EDF$AB_ACCESS_TABLE_STA),
5011                                    FALSE,
5012                                    0
5013                                    );
5014
5015                    END;        { ACCESS }
```

```
5017            (* Here starts the comment to exclude ACLS *)
5018            (*          ACL :
5019
5020                        BEGIN
5021
5022                                IF FULL_CHOICE THEN
5023
5024                                    BEGIN
5025
5026                                        CLEAR (IF_FULL_PROMPT);
5027
5028                                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5029
5030                                        BEGIN
5031
5032                                            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5033                                            ' Legal ACL',SEC_ATTR,
5034                                            ANSI_RESET,CRLF,
5035                                            CRLF,SHIFT,
5036                                            'ENTRY                  string',
5037                                            CRLF);
5038            *)
5039            (*                              END*){ IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5040            (*
5041                                        ELSE
5042
5043                                            WRITELN (SHIFT,QUES_HINT);
5044
5045                                    END
5046
5047                                ELSE
5048
5049                                    BEGIN
5050
5051                                        CLEAR (IF_FULL_PROMPT);
5052
5053                                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5054
5055                                        BEGIN
5056
5057                                            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5058                                            ' Current ACL',SEC_ATTR,
5059                                            ANSI_RESET,CRLF);
5060            *)
5061                                            { +
5062                                            Setup to display definition on the terminal.
5063                                            - }
5064            (*                              OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5065                                                    RECORD_LENGTH := 252);
5066                                            REWRITE (FDL_DEST);
5067
5068                                            SHOW_PRIMARY_SECTION (TEST);
5069
5070                                            CLOSE (FDL_DEST);
5071            *)
5072            (*                              END*){ IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5073            (*
```

```
5074                                    ELSE
5075
5076                                        WRITELN (SHIFT,QUES_HINT);
5077
5078                                END;*)  { EXTANT_ONLY }
5079
5080            { +
5081                THIS CAN BE OPTIMIZED IN THE FUTURE - GIVEN THAT THE ACL PRIMARY HAS ONLY
5082                ONE KIND OF SECONDARY:   "ENTRY"
5083            - }
5084            (*
5085                            WRITE (SHIFT,'Enter ACL Attribute                (Keyword)',
5086                            ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5087                            PARSE_INPUT (
5088                                    IADDRESS (EDF$AB_ACL_TABLE_KEY),
5089                                    IADDRESS (EDF$AB_ACL_TABLE_STA),
5090                                    FALSE,
5091                                    0
5092                                    );
5093
5094                        END;*)      { ACL }
5095
```

```
5097                    ANALYSIS_OF_AREA :
5098
5099                        INPUT_VALUE     := 0;   { DUMMY_SECONDARY$ }
5100
5101                    ANALYSIS_OF_KEY :
5102
5103                        INPUT_VALUE     := 0;   { DUMMY_SECONDARY$ }
```

```
5105                    AREA :
5106
5107                    BEGIN
5108
5109                            IF FULL_CHOICE THEN
5110
5111                                BEGIN
5112
5113                                    CLEAR (IF_FULL_PROMPT);
5114
5115                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5116
5117                                    BEGIN
5118
5119                                        WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5120                                        ' Legal AREA ',ACTIVE_AREA:NUM_LEN(ACTIVE_AREA),
5121                                        SEC_ATTR,
5122                                        ANSI_RESET,CRLF,
5123                                        CRLF_SHIFT,
5124                                        'ALLOCATION              number',
5125                                        CRLF_SHIFT,
5126                                        'BEST_TRY_CONTIGUOUS     yes/no',
5127                                        CRLF_SHIFT,
5128                                        'BUCKET_SIZE             number',
5129                                        CRLF_SHIFT,
5130                                        'CONTIGUOUS              yes/no',
5131                                        CRLF_SHIFT,
5132                                        'EXACT_POSITIONING       yes/no',
5133                                        CRLF_SHIFT,
5134                                        'EXTENSION               number',
5135                                        CRLF_SHIFT,
5136                                        'POSITION    qualifier   number',
5137                                        CRLF_SHIFT,
5138                                        'VOLUME                  number',
5139                                        CRLF);
5140
5141                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5142
5143                                    ELSE
5144
5145                                        WRITELN (SHIFT,QUES_HINT);
5146
5147                                END
5148
5149                            ELSE
5150
5151                                BEGIN
5152
5153                                    CLEAR (IF_FULL_PROMPT);
5154
5155                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5156
5157                                    BEGIN
5158
5159                                        WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5160                                        ' Current AREA ',
5161                                        ACTIVE_AREA:NUM_LEN(ACTIVE_AREA),
```

```
5162                        SEC_ATTR,
5163                        ANSI_RESET,CRLF);
5164
5165                        { +
5166                        Setup to display definition on the terminal.
5167                        - }
5168                        OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5169                                 RECORD_LENGTH := 252);
5170                        REWRITE (FDL_DEST);
5171
5172                        SHOW_PRIMARY_SECTION (TEST);
5173
5174                        CLOSE (FDL_DEST);
5175
5176                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5177
5178                    ELSE
5179
5180                        WRITELN (SHIFT,QUES_HINT);
5181
5182            END;    { EXTANT_ONLY }
5183
5184        WRITE (SHIFT,'Enter AREA ',ACTIVE_AREA:NUM_LEN(ACTIVE_AREA),
5185        ' Attribute             (Keyword)',
5186        ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5187        PARSE_INPUT (
5188                    IADDRESS (EDF$AB_AREA_TABLE_KEY),
5189                    IADDRESS (EDF$AB_AREA_TABLE_STA),
5190                    FALSE,
5191                    0
5192                    );
5193
5194            END;        { AREA }
```

```
5196                    CONNECT :
5197
5198                    BEGIN
5199
5200                            IF FULL_CHOICE THEN
5201
5202                                BEGIN
5203
5204                                    CLEAR (IF_FULL_PROMPT);
5205
5206                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5207
5208                                    BEGIN
5209
5210            WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5211            ' Legal CONNECT',SEC_ATTR,
5212            ANSI_RESET,CRLF,
5213            CRLF_SHIFT,
5214            'ASYNCHRONOUS           yes/no  NOLOCK                  yes/no',
5215            CRLF_SHIFT,
5216            'BLOCK_IO               yes/no  NONEXISTENT_RECORD      yes/no',
5217            CRLF_SHIFT,
5218            'BUCKET_CODE            number  READ_AHEAD              yes/no',
5219            CRLF_SHIFT,
5220            'CONTEXT                        number  READ_REGARDLESS        yes/no',
5221            CRLF_SHIFT,
5222            'END_OF_FILE            yes/no  TIMEOUT_ENABLE          yes/no',
5223            CRLF_SHIFT,
5224            'FAST_DELETE            yes/no  TIMEOUT_PERIOD          number',
5225            CRLF_SHIFT,
5226            'FILE_BUCKETS           yes/no  TRUNCATE_ON_PUT         yes/no',
5227            CRLF_SHIFT,
5228            'KEY_GREATER_EQUAL      yes/no  TT_CANCEL_CONTROL_O     yes/no',
5229            CRLF_SHIFT,
5230            'KEY_GREATER_THAN       yes/no  TT_PROMPT               yes/no',
5231            CRLF_SHIFT,
5232            'KEY_LIMIT              yes/no  TT_PURGE_TYPE_AHEAD     yes/no',
5233            CRLF_SHIFT,
5234            'KEY_OF_REFERENCE       number  TT_READ_NOECHO          yes/no',
5235            CRLF_SHIFT,
5236            'LOCATE_MODE            yes/no  TT_READ_NOFILTER        yes/no',
5237            CRLF_SHIFT,
5238            'LOCK_ON_READ           yes/no  TT_UPCASE_INPUT         yes/no',
5239            CRLF_SHIFT,
5240            'LOCK_ON_WRITE          yes/no  UPDATE_IF               yes/no',
5241            CRLF_SHIFT,
5242            'MANUAL_UNLOCKING       yes/no  WAIT_FOR_RECORD         yes/no',
5243            CRLF_SHIFT,
5244            'MULTIBLOCK_COUNT       number  WRITE_BEHIND            yes/no',
5245            CRLF_SHIFT,
5246            'MULTIBUFFER_COUNT      number',
5247            CRLF
5248            );
5249
5250                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5251
5252                                    ELSE
```

```
5253
5254                            WRITELN (SHIFT,QUES_HINT);
5255
5256                        END
5257
5258                    ELSE
5259
5260                        BEGIN
5261
5262                            CLEAR (IF_FULL_PROMPT);
5263
5264                            IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5265
5266                            BEGIN
5267
5268                                WRITELN (SHIFT,'                ',ANSI_REVERSE,
5269                                ' Current CONNECT',SEC_ATTR,
5270                                ANSI_RESET,CRLF);
5271
5272                                { +
5273                                Setup to display definition on the terminal.
5274                                - }
5275                                OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5276                                        RECORD_LENGTH := 252);
5277                                REWRITE (FDL_DEST);
5278
5279                                SHOW_PRIMARY_SECTION (TEST);
5280
5281                                CLOSE (FDL_DEST);
5282
5283                            END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5284
5285                            ELSE
5286
5287                                WRITELN (SHIFT,QUES_HINT);
5288
5289                        END;    { EXTANT_ONLY }
5290
5291                    WRITE (SHIFT,'Enter CONNECT Attribute          (Keyword)',
5292                    ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5293                    PARSE_INPUT (
5294                                IADDRESS (EDF$AB_CONNECT_TABLE_KEY),
5295                                IADDRESS (EDF$AB_CONNECT_TABLE_STA),
5296                                FALSE,
5297                                0
5298                                );
5299
5300                END;        { CONNECT }
```

```
5302                    DATE :
5303
5304                    BEGIN
5305
5306                            IF FULL_CHOICE THEN
5307
5308                                BEGIN
5309
5310                                    CLEAR (IF_FULL_PROMPT);
5311
5312                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5313
5314                                    BEGIN
5315
5316                                        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5317                                        ' Legal DATE',SEC_ATTR,
5318                                        ANSI_RESET,CRLF,
5319                                        CRLF,SHIFT,
5320                                        'BACKUP                 string',
5321                                        CRLF,SHIFT,
5322                                        'CREATION               string',
5323                                        CRLF,SHIFT,
5324                                        'EXPIRATION             string',
5325                                        CRLF,SHIFT,
5326                                        'REVISION               string',
5327                                        CRLF);
5328
5329                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5330
5331                                    ELSE
5332
5333                                        WRITELN (SHIFT,QUES_HINT);
5334
5335                                END
5336
5337                        ELSE
5338
5339                                BEGIN
5340
5341                                    CLEAR (IF_FULL_PROMPT);
5342
5343                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5344
5345                                    BEGIN
5346
5347                                        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5348                                        ' Current DATE',SEC_ATTR,
5349                                        ANSI_RESET,CRLF);
5350
5351                                        { +
5352                                        Setup to display definition on the terminal.
5353                                        - }
5354                                        OPEN    (FDL_DEST,SYS$OUTPUT,NAME,NEW,
5355                                                RECORD_LENGTH := 252);
5356                                        REWRITE (FDL_DEST);
5357
5358                                        SHOW_PRIMARY_SECTION (TEST);
```

```
5359
5360                                    CLOSE (FDL_DEST);
5361
5362                                END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5363
5364                                ELSE
5365
5366                                    WRITELN (SHIFT,QUES_HINT);
5367
5368                            END;    { EXTANT_ONLY }
5369
5370                        WRITE (SHIFT,'Enter DATE Attribute                (Keyword)',
5371                        ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5372                        PARSE_INPUT (
5373                                    IADDRESS (EDF$AB_DATE_TABLE_KEY),
5374                                    IADDRESS (EDF$AB_DATE_TABLE_STA),
5375                                    FALSE,
5376                                    0
5377                                    );
5378
5379                    END;        { DATE }
```

```
5381                    FILE$ :
5382
5383                    BEGIN
5384
5385                            IF FULL_CHOICE THEN
5386
5387                                BEGIN
5388
5389                                    CLEAR (IF_FULL_PROMPT);
5390
5391                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5392
5393                                    BEGIN
5394
5395            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5396            ' Legal FILE',SEC_ATTR,
5397            ANSI_RESET,CRLF,CRLF_SHIFT,
5398            'ALLOCATION             number  MT_PROTECTION           char/num',
5399            CRLF_SHIFT,
5400            'BEST_TRY_CONTIGUOUS    yes/no  NAME                    string',
5401            CRLF_SHIFT,
5402            'BUCKET_SIZE            number  NOBACKUP                yes/no',
5403            CRLF_SHIFT,
5404            'CLUSTER_SIZE           number  NON_FILE_STRUCTURED     yes/no',
5405            CRLF_SHIFT,
5406            'CONTEXT                        number  ORGANIZATION           keyword',
5407            CRLF_SHIFT,
5408            'CONTIGUOUS             yes/no  OUTPUT_FILE_PARSE      yes/no',
5409            CRLF_SHIFT,
5410            'CREATE_IF              yes/no  OWNER                   uic',
5411            CRLF_SHIFT,
5412            'DEFAULT_NAME           string  PRINT_ON_CLOSE          yes/no',
5413            CRLF_SHIFT,
5414            'DEFERRED_WRITE         yes/no  PROTECTION              yes/no',
5415            CRLF_SHIFT,
5416            'DELETE_ON_CLOSE                yes/no  READ_CHECK              yes/no',
5417            CRLF_SHIFT,
5418            'DIRECTORY_ENTRY                yes/no  REVISION                number',
5419            CRLF_SHIFT,
5420            'EXTENSION              number  SEQUENTIAL_ONLY         yes/no',
5421            CRLF_SHIFT,
5422            'GLOBAL_BUFFER_COUNT    number  SUBMIT_ON_CLOSE         yes/no',
5423            CRLF_SHIFT,
5424            'MAX_RECORD_NUMBER      number  SUPERSEDE               yes/no',
5425            CRLF_SHIFT,
5426            'MAXIMIZE_VERSION       yes/no  TEMPORARY               yes/no',
5427            CRLF_SHIFT,
5428            'MT_BLOCK_SIZE          number  TRUNCATE_ON_CLOSE       yes/no',
5429            CRLF_SHIFT,
5430            'MT_CLOSE_REWIND                yes/no  USER_FILE_OPEN         yes/no',
5431            CRLF_SHIFT,
5432            'MT_CURRENT_POSITION    yes/no  WINDOW_SIZE             number',
5433            CRLF_SHIFT,
5434            'MT_NOT_EOF             yes/no  WRITE_CHECK             yes/no',
5435            CRLF
5436            );
5437
```

```
5438                        END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5439
5440                        ELSE
5441
5442                            WRITELN (SHIFT,QUES_HINT);
5443
5444                    END
5445
5446                ELSE
5447
5448                    BEGIN
5449
5450                        CLEAR (IF_FULL_PROMPT);
5451
5452                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5453
5454                        BEGIN
5455
5456                            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5457                            ' Current FILE',SEC_ATTR,
5458                            ANSI_RESET,CRLF);
5459
5460                            { +
5461                            Setup to display definition on the terminal.
5462                            - }
5463                            OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5464                                    RECORD_LENGTH := 252);
5465                            REWRITE (FDL_DEST);
5466
5467                            SHOW_PRIMARY_SECTION (TEST);
5468
5469                            CLOSE (FDL_DEST);
5470
5471                        END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5472
5473                        ELSE
5474
5475                            WRITELN (SHIFT,QUES_HINT);
5476
5477                    END;    { EXTANT_ONLY }
5478
5479                WRITE (SHIFT,'Enter FILE Attribute              (Keyword)',
5480                ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5481                PARSE_INPUT (
5482                            IADDRESS (EDF$AB_FILE_TABLE_KEY),
5483                            IADDRESS (EDF$AB_FILE_TABLE_STA),
5484                            FALSE,
5485                            0
5486                            );
5487
5488            END;        { FILE }
```

```
5490                    JOURNAL :
5491
5492                    BEGIN
5493
5494                            IF FULL_CHOICE THEN
5495
5496                                BEGIN
5497
5498                                    CLEAR (IF_FULL_PROMPT);
5499
5500                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5501
5502                                    BEGIN
5503
5504                                        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5505                                        ' Legal JOURNAL',SEC_ATTR,
5506                                        ANSI_RESET,CRLF,
5507                                        CRLF_SHIFT,
5508                                        'AFTER_IMAGE            yes/no',
5509                                        CRLF_SHIFT,
5510                                        'AFTER_NAME             string',
5511                                        CRLF_SHIFT,
5512                                        'AUDIT_TRAIL            yes/no',
5513                                        CRLF_SHIFT,
5514                                        'AUDIT_NAME             string',
5515                                        CRLF_SHIFT,
5516                                        'BEFORE_IMAGE           yes/no',
5517                                        CRLF_SHIFT,
5518                                        'BEFORE_NAME            string',
5519                                        CRLF_SHIFT,
5520                                        'RECOVERY_UNIT          keyword',
5521                                        CRLF);
5522
5523                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5524
5525                                    ELSE
5526
5527                                    BEGIN
5528
5529                                        WRITELN (SHIFT,QUES_HINT);
5530
5531                                    END;
5532
5533                                END
5534
5535                            ELSE
5536
5537                                BEGIN
5538
5539                                    CLEAR (IF_FULL_PROMPT);
5540
5541                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5542
5543                                    BEGIN
5544
5545                                        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5546                                        ' Current JOURNAL',SEC_ATTR,
```

```
5547                        ANSI_RESET,CRLF);
5548
5549                        { +
5550                        Setup to display definition on the terminal.
5551                        - }
5552                        OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5553                                 RECORD_LENGTH := 252);
5554                        REWRITE (FDL_DEST);
5555
5556                        SHOW_PRIMARY_SECTION (TEST);
5557
5558                        CLOSE (FDL_DEST);
5559
5560                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5561
5562                    ELSE
5563
5564                        WRITELN (SHIFT,QUES_HINT);
5565
5566            END;    { EXTANT_ONLY }
5567
5568        WRITE (SHIFT,'Enter JOURNAL Attribute          (Keyword)',
5569        ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5570        PARSE_INPUT (
5571                    IADDRESS (EDF$AB_JOURNAL_TABLE_KEY),
5572                    IADDRESS (EDF$AB_JOURNAL_TABLE_STA),
5573                    FALSE,
5574                    0
5575                    );
5576
5577                END;        { JOURNAL }
```

```
5579                    KEY :
5580
5581                    BEGIN
5582
5583                            IF FULL_CHOICE THEN
5584
5585                                BEGIN
5586
5587                                    CLEAR (IF_FULL_PROMPT);
5588
5589                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5590
5591                                    BEGIN
5592
5593        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5594        ' Legal KEY',
5595        IDATA[EDF$K_ACTIVE_KEY]:3,
5596        SEC_ATTR,
5597        ANSI_RESET,CRLF,
5598        CRLF_SHIFT,
5599        'CHANGES                        yes/no  LEVEL1_INDEX_AREA       number',
5600        CRLF_SHIFT,
5601        'DATA_AREA              number  NAME                    string',
5602        CRLF_SHIFT,
5603        'DATA_FILL              number  NULL_KEY                yes/no',
5604        CRLF_SHIFT,
5605        'DATA_KEY_COMPRESSION   yes/no  NULL_VALUE              char/num',
5606        CRLF_SHIFT,
5607        'DATA_RECORD_COMPRESSION        yes/no  POSITION                number',
5608        CRLF_SHIFT,
5609        'DUPLICATES             yes/no  PROLOG                  number',
5610        CRLF_SHIFT,
5611        'INDEX_AREA             number  TYPE                    keyword',
5612        CRLF_SHIFT,
5613        'INDEX_COMPRESSION      yes/no  SEGn_LENGTH             number',
5614        CRLF_SHIFT,
5615        'INDEX_FILL             number  SEGn_POSITION           number',
5616        CRLF_SHIFT,
5617        'LENGTH                 number'{        SEGn_TYPE       keyword},
5618        CRLF);
5619
5620                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5621
5622                                    ELSE
5623
5624                                        WRITELN (SHIFT,QUES_HINT);
5625
5626                                END
5627
5628                            ELSE
5629
5630                                BEGIN
5631
5632                                    CLEAR (IF_FULL_PROMPT);
5633
5634                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5635
```

```
5636                    BEGIN
5637
5638                        WRITELN (SHIFT,'                ',ANSI_REVERSE,
5639                        ' Current KEY',
5640                        IDATA[EDF$K_ACTIVE_KEY]:3,
5641                        SEC_ATTR,
5642                        ANSI_RESET,CRLF);
5643
5644                        { +
5645                        Setup to display definition on the terminal.
5646                        - }
5647                        OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5648                                RECORD_LENGTH := 252);
5649                        REWRITE (FDL_DEST);
5650
5651                        SHOW_PRIMARY_SECTION (TEST);
5652
5653                        CLOSE (FDL_DEST);
5654
5655                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5656
5657                    ELSE
5658
5659                        WRITELN (SHIFT,QUES_HINT);
5660
5661                END;    { EXTANT_ONLY }
5662
5663                WRITE (SHIFT,'Enter KEY',
5664                IDATA[EDF$K_ACTIVE_KEY]:3,
5665                ' Attribute             (Keyword)',
5666                ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5667                PARSE_INPUT (
5668                            IADDRESS (EDF$AB_KEY_TABLE_KEY),
5669                            IADDRESS (EDF$AB_KEY_TABLE_STA),
5670                            FALSE,
5671                            0
5672                            );
5673
5674            END;        { KEY }
```

```
5676                    RECORD$ :
5677
5678                    BEGIN
5679
5680                            IF FULL_CHOICE THEN
5681
5682                                BEGIN
5683
5684                                    CLEAR (IF_FULL_PROMPT);
5685
5686                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5687
5688                                    BEGIN
5689
5690                                        WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5691                                        ' Legal RECORD',SEC_ATTR,
5692                                        ANSI_RESET,CRLF,
5693                                        CRLF_SHIFT,
5694                                        'BLOCK_SPAN              yes/no',
5695                                        CRLF_SHIFT,
5696                                        'CARRIAGE_CONTROL        keyword',
5697                                        CRLF_SHIFT,
5698                                        'CONTROL_FIELD_SIZE      number',
5699                                        CRLF_SHIFT,
5700                                        'FORMAT                  keyword',
5701                                        CRLF_SHIFT,
5702                                        'SIZE                    number',
5703                                        CRLF);
5704
5705                                    END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5706                                    ELSE
5707
5708                                    BEGIN
5709
5710                                        WRITELN (SHIFT,QUES_HINT);
5711
5712                                    END;
5713
5714                                END
5715
5716                            ELSE
5717
5718                                BEGIN
5719
5720                                    CLEAR (IF_FULL_PROMPT);
5721
5722                                    IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5723
5724                                    BEGIN
5725
5726                                        WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5727                                        ' Current RECORD',SEC_ATTR,
5728                                        ANSI_RESET,CRLF);
5729
5730                                        { +
5731                                        Setup to display definition on the terminal.
5732
```

```
5733                                    - }
5734                                    OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5735                                            RECORD_LENGTH := 252);
5736                                    REWRITE (FDL_DEST);
5737
5738                                    SHOW_PRIMARY_SECTION (TEST);
5739
5740                                    CLOSE (FDL_DEST);
5741
5742                                END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5743
5744                                ELSE
5745
5746                                    WRITELN (SHIFT,QUES_HINT);
5747
5748                            END;    { EXTANT_ONLY }
5749
5750                        WRITE (SHIFT,'Enter RECORD Attribute            (Keyword)',
5751                        ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5752                        PARSE_INPUT (
5753                                    IADDRESS (EDF$AB_RECORD_TABLE_KEY),
5754                                    IADDRESS (EDF$AB_RECORD_TABLE_STA),
5755                                    FALSE,
5756                                    0
5757                                    );
5758
5759                END;        { RECORD }
```

```
5761            SHARING :
5762
5763            BEGIN
5764
5765                    IF FULL_CHOICE THEN
5766
5767                        BEGIN
5768
5769                            CLEAR (IF_FULL_PROMPT);
5770
5771                            IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5772
5773                            BEGIN
5774
5775                                WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5776                                ' Legal SHARING',SEC_ATTR,
5777                                ANSI_RESET,CRLF,
5778                                CRLF_SHIFT,
5779                                'DELETE                  yes/no',
5780                                CRLF_SHIFT,
5781                                'GET                     yes/no',
5782                                CRLF_SHIFT,
5783                                'MULTISTREAM             yes/no',
5784                                CRLF_SHIFT,
5785                                'PROHIBIT                yes/no',
5786                                CRLF_SHIFT,
5787                                'PUT                     yes/no',
5788                                CRLF_SHIFT,
5789                                'UPDATE                  yes/no',
5790                                CRLF_SHIFT,
5791                                'USER_INTERLOCK          yes/no',
5792                                CRLF);
5793
5794                            END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5795
5796                            ELSE
5797
5798                                WRITELN (SHIFT,QUES_HINT);
5799
5800                        END
5801
5802                ELSE
5803
5804                        BEGIN
5805
5806                            CLEAR (IF_FULL_PROMPT);
5807
5808                            IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5809
5810                            BEGIN
5811
5812                                WRITELN (SHIFT,'                    ',ANSI_REVERSE,
5813                                ' Current SHARING',SEC_ATTR,
5814                                ANSI_RESET,CRLF);
5815
5816                                { *
5817                                Setup to display definition on the terminal.
```

```
5818                            - }
5819                            OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5820                                    RECORD_LENGTH := 252);
5821                            REWRITE (FDL_DEST);
5822
5823                            SHOW_PRIMARY_SECTION (TEST);
5824
5825                            CLOSE (FDL_DEST);
5826
5827                        END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5828
5829                        ELSE
5830
5831                            WRITELN (SHIFT,QUES_HINT);
5832
5833                    END;    { EXTANT_ONLY }
5834
5835            WRITE (SHIFT,'Enter SHARING Attribute           (Keyword)',
5836            ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5837            PARSE_INPUT (
5838                        IADDRESS (EDF$AB_SHARING_TABLE_KEY),
5839                        IADDRESS (EDF$AB_SHARING_TABLE_STA),
5840                        FALSE,
5841                        0
5842                        );
5843
5844        END;        { SHARING }
```

```
5846        SYSTEM :
5847
5848        BEGIN
5849
5850                IF FULL_CHOICE THEN
5851
5852                    BEGIN
5853
5854                        CLEAR (IF_FULL_PROMPT);
5855
5856                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5857
5858                        BEGIN
5859
5860                            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5861                            ' Legal SYSTEM',SEC_ATTR,
5862                            ANSI_RESET,CRLF,
5863                            CRLF_SHIFT,
5864                            'DEVICE                 string',
5865                            CRLF_SHIFT,
5866                            'SOURCE                 keyword',
5867                            CRLF_SHIFT,
5868                            'TARGET                 keyword',
5869                            CRLF);
5870
5871                        END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5872
5873                        ELSE
5874
5875                            WRITELN (SHIFT,QUES_HINT);
5876
5877                    END
5878
5879                ELSE
5880
5881                    BEGIN
5882
5883                        CLEAR (IF_FULL_PROMPT);
5884
5885                        IF FULL_PROMPT OR TEMP_FULL_PROMPT THEN
5886
5887                        BEGIN
5888
5889                            WRITELN (SHIFT,'                ',ANSI_REVERSE,
5890                            ' Current SYSTEM',SEC_ATTR,
5891                            ANSI_RESET,CRLF);
5892
5893                            { +
5894                            Setup to display definition on the terminal.
5895                            - }
5896                            OPEN    (FDL_DEST,SYS$OUTPUT_NAME,NEW,
5897                                    RECORD_LENGTH := 252);
5898                            REWRITE (FDL_DEST);
5899
5900                            SHOW_PRIMARY_SECTION (TEST);
5901
5902                            CLOSE (FDL_DEST);
```

```
5903
5904                        END { IF FULL_PROMPT OR TEMP_FULL_PROMPT TRUE }
5905
5906                        ELSE
5907
5908                            WRITELN (SHIFT,QUES_HINT);
5909
5910                    END;    { EXTANT_ONLY }
5911
5912                WRITE (SHIFT,'Enter SYSTEM Attribute            (Keyword)',
5913                ANSI_REVERSE,'[-]',ANSI_RESET,' : ');
5914                PARSE_INPUT (
5915                            IADDRESS (EDF$AB_SYSTEM_TABLE_KEY),
5916                            IADDRESS (EDF$AB_SYSTEM_TABLE_STA),
5917                            FALSE,
5918                            0
5919                            );
5920
5921            END;        { SYSTEM }
```

```
5923        OTHERWISE
5924
5925            { NULL-STATEMENT } ;
5926
5927        END;    { CASE }
5928
5929        IF TEST.PRIMARY <> TITLE THEN
5930
5931            TEST.OBJECT_TYPE    := SEC;
5932
5933        TEST.SECONDARY  := INPUT_VALUE::SECONDARY_TYPE;
5934
5935        { +
5936        Get the secondary number if there was one, it's inited to 0;
5937        only SEGn_LENGTH,_POSITION,_TYPE set it.
5938        Force seg_type to be last.
5939        - }
5940        IF TEST.SECONDARY = SEG_TYPE THEN
5941
5942            TEST.SECNUM := 7
5943
5944        ELSE
5945
5946            TEST.SECNUM := EDF$GL_SECNUM;
5947
5948        IF (
5949        (TEST.SECNUM < 0)
5950        OR
5951        (TEST.SECNUM > 7)
5952        ) THEN
5953
5954            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
5955
5956        { +
5957        If we're only to ask for what exists, then make sure this does.
5958        - }
5959        IF NOT FULL_CHOICE THEN
5960
5961        BEGIN
5962
5963            DEF_CURRENT         := DEF_HEAD;
5964
5965            REPEAT
5966
5967                IF NOT CURRENT_EQ_TEST(TEST,TRUE) THEN
5968
5969                    INCR_CURRENT;
5970
5971            UNTIL (CURRENT_EQ_TEST(TEST,TRUE) OR (DEF_CURRENT^.FORE = NIL));
5972
5973            IF DEF_CURRENT <> NIL THEN
5974
5975            BEGIN
5976
5977                IF NOT CURRENT_EQ_TEST(TEST,TRUE) THEN
5978
5979                    LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
```

```
5980
5981                    END
5982
5983                    ELSE
5984
5985                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
5986
5987                END;    { IF DISPLAY = EXTANT_ONLY }
5988
5989                { +
5990                Make sure this is true for only one cycle.
5991                - }
5992                TEMP_FULL_PROMPT        := FALSE;
5993
5994            END;        { THE_QUESTION }
5995
5996        BEGIN
5997
5998            { +
5999            Keep at it until the user gets it right.
6000            - }
6001            REPEAT
6002
6003                THE_QUESTION;
6004
6005            UNTIL NOT SYS$INPUT_ERROR;
6006
6007            STR$FREE1_DX (INPUT_DESC);
6008
6009        END;    { ASK_TEST_SECONDARY }
```

```
6011        { ++
6012
6013        ASK_TEST_SECONDARY_VALUE -- Input the user's value for the secondary.
6014
6015        This routine queries the user about his secondary.
6016
6017        CALLING SEQUENCE:
6018
6019        ASK_TEST_SECONDARY_VALUE;
6020
6021        INPUT PARAMETERS:
6022
6023        none
6024
6025        IMPLICIT INPUTS:
6026
6027        SYS$INPUT:
6028
6029        OUTPUT PARAMETERS:
6030
6031        none
6032
6033        IMPLICIT OUTPUTS:
6034
6035        SYS$INPUT_ERROR
6036        SYS$OUTPUT:
6037
6038        ROUTINES CALLED:
6039
6040        ESTABLISH
6041
6042        ROUTINE VALUE:
6043
6044        none
6045
6046        SIGNALS:
6047
6048        none
6049
6050        SIDE EFFECTS:
6051
6052        none
6053
6054        -- }
```

```
6056        PROCEDURE ASK_TEST_SECONDARY_VALUE;
6057
6058        VAR
6059            I                   : INTEGER;
6060            TEMP_INT            : INTEGER;
6061            TEMP_DESC           : DESCRIPTOR;
6062            TEMP_MAX            : INTEGER;
6063            TEMP_STRING255      : STRING255;
6064
6065            PROCEDURE THE_QUESTION;
6066
6067            BEGIN
6068
6069                { +
6070                Set up the condition handler to catch typing errors.
6071                - }
6072                SYS$INPUT_ERROR := FALSE;
6073                ESTABLISH (SYS$INPUT_COND_HANDLER);
6074
6075                IF TEMP_FULL_PROMPT THEN
6076
6077                    WRITELN (SHIFT,
6078                    'The value entered will be put into the Definition.');
6079
6080                { +
6081                Pop the question.
6082                - }
6083                IF TEST.PRIMARY = DATE THEN
6084
6085                    WRITE (CRLF_SHIFT,
6086                    '(dd-mmm-yyyy hh:mm:ss.cc)');
6087
6088            IF TEST.SECONDARY = POSITION$ THEN
6089
6090                WRITE (CRLF_SHIFT,
6091                '(Any_cylinder Cylinder File_ID File_name',
6092                CRLF_SHIFT,' Logical None Virtual)',CRLF_SHIFT,
6093                'Enter POSITION qualifier   (')
6094
6095            ELSE IF NOT SEC_TYPE[TEST.SECONDARY].QUAL THEN
6096
6097                WRITE (CRLF_SHIFT,'Enter value for this Secondary   (');
6098
6099            IF SEC_TYPE[TEST.SECONDARY].QUAL THEN
6100
6101            BEGIN
6102
6103                CASE TEST.SECONDARY OF
6104
6105                    ORGANIZATION :
6106
6107                        WRITE (CRLF_SHIFT,
6108                        '(Indexed Relative Sequential)');
6109
6110                    SOURCE,
6111                    TARGET :
6112
```

```
6113                        WRITE (CRLF_SHIFT,
6114                        '(IAS RSTS/E RSX-11M RSX-11M-PLUS RT-11 VAX/VMS)');
6115
6116                        RECOVERY_UNIT :
6117
6118                            WRITE (CRLF_SHIFT,
6119                      '(If_in_recovery_unit Necessary_to_write Never_RU_journal None)');
6120
6121                        CARRIAGE_CONTROL :
6122
6123                            WRITE (CRLF_SHIFT,
6124                            '(Carriage_return FORTRAN None Print)');
6125
6126                        FORMAT :
6127
6128                            WRITE (CRLF_SHIFT,
6129                            '(Fixed Stream Stream_CR Stream_LF',CRLF_SHIFT,
6130                            ' Undefined Variable VFC)');
6131
6132                        SEG_TYPE :
6133
6134                            WRITE (CRLF_SHIFT,
6135                            '(Bin2 Bin4 Bin8 Decimal Int2 Int4 Int8 String)');
6136
6137                    OTHERWISE
6138
6139                        { NULL-STATEMENT } ;
6140
6141                    END;        { CASE }
6142
6143                    WRITE (CRLF_SHIFT,'Enter value for this Secondary   (Keyword)',
6144                    ANSI_REVERSE,'[-]',ANSI_RESET,'     : ');
6145
6146                    CASE TEST.SECONDARY OF
6147
6148                        ORGANIZATION :
6149
6150                        PARSE_INPUT (
6151                                    IADDRESS (EDF$AB_ORG_TABLE_KEY),
6152                                    IADDRESS (EDF$AB_ORG_TABLE_STA),
6153                                    FALSE,
6154                                    0
6155                                    );
6156
6157                        SOURCE,
6158                        TARGET :
6159
6160                        PARSE_INPUT (
6161                                    IADDRESS (EDF$AB_SOURCE_TABLE_KEY),
6162                                    IADDRESS (EDF$AB_SOURCE_TABLE_STA),
6163                                    FALSE,
6164                                    0
6165                                    );
6166
6167                        RECOVERY_UNIT :
6168
6169                        PARSE_INPUT (
```

```
6170                        IADDRESS (EDF$AB_RU_TABLE_KEY),
6171                        IADDRESS (EDF$AB_RU_TABLE_STA),
6172                        FALSE,
6173                        0
6174                        );
6175
6176                    CARRIAGE_CONTROL :
6177
6178                    PARSE_INPUT (
6179                        IADDRESS (EDF$AB_CARR_TABLE_KEY),
6180                        IADDRESS (EDF$AB_CARR_TABLE_STA),
6181                        FALSE,
6182                        0
6183                        );
6184
6185                    FORMAT :
6186
6187                    PARSE_INPUT (
6188                        IADDRESS (EDF$AB_FORMAT_TABLE_KEY),
6189                        IADDRESS (EDF$AB_FORMAT_TABLE_STA),
6190                        FALSE,
6191                        0
6192                        );
6193
6194                    SEG_TYPE :
6195
6196                    PARSE_INPUT (
6197                        IADDRESS (EDF$AB_TYPE_TABLE_KEY),
6198                        IADDRESS (EDF$AB_TYPE_TABLE_STA),
6199                        FALSE,
6200                        0
6201                        );
6202
6203                OTHERWISE
6204
6205                    { NULL-STATEMENT } ;
6206
6207                END;        { CASE }
6208
6209                TEST.QUALIFIER      := INPUT_VALUE;
6210
6211            END;    { IF QUALIFIER_VALUED }
6212
6213            IF SEC_TYPE[TEST.SECONDARY].NUM THEN
6214
6215            BEGIN
6216
6217                TEMP_MAX    := SECONDARY_MAX[TEST.SECONDARY];
6218
6219                IF (TEST.SECONDARY IN
6220
6221                [ DATA_KEY_COMPRESSION,
6222                DATA_RECORD_COMPRESSION,
6223                INDEX_COMPRESSION ]
6224
6225                ) THEN
6226
```

```
6227                        WRITE ('Abs<100)')
6228
6229                    ELSE IF TEMP_MAX = EDF$C_1GIGA THEN
6230
6231                        WRITE ('0-1Giga)')
6232
6233                    ELSE
6234
6235                        WRITE ('0-',TEMP_MAX:NUM_LEN(TEMP_MAX),')');
6236
6237                    WRITE (ANSI_REVERSE,'[-]',ANSI_RESET);
6238
6239                    IF (
6240                    (NUM_LEN(TEMP_MAX) > 8)
6241                    AND
6242                    (TEMP_MAX <> EDF$C_1GIGA)
6243                    ) THEN
6244
6245                        WRITE (' : ')
6246
6247                    ELSE
6248
6249                        WRITE ('         : ');
6250
6251                    NUMBER_INPUT (TEST.NUMBER,FALSE,0);
6252
6253                    IF (TEST.SECONDARY IN
6254
6255                    [ DATA_KEY_COMPRESSION,
6256                    DATA_RECORD_COMPRESSION,
6257                    INDEX_COMPRESSION ]
6258
6259                    ) THEN
6260
6261                    BEGIN
6262
6263                        IF (
6264                        ((VDATA[EDF$K_PROLOGUE_VERSION])
6265                        AND (IDATA[EDF$K_PROLOGUE_VERSION] < 3))
6266                        AND
6267                        (TEST.NUMBER <> 0)
6268                        ) THEN
6269
6270                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6271
6272                        IF (
6273                        (TEST.NUMBER < -TEMP_MAX)
6274                        OR
6275                        (TEST.NUMBER > TEMP_MAX)
6276                        ) THEN
6277
6278                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6279
6280                    END
6281
6282                    ELSE IF TEST.SECONDARY = CONTROL_FIELD_SIZE THEN
6283
```

```
6284                    BEGIN
6285
6286                        IF (
6287                        (TEST.NUMBER < 1)
6288                        OR
6289                        (TEST.NUMBER > TEMP_MAX)
6290                        ) THEN
6291
6292                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6293
6294                    END
6295
6296                    ELSE
6297
6298                    BEGIN
6299
6300                        IF (
6301                        (TEST.NUMBER < 0)
6302                        OR
6303                        (TEST.NUMBER > TEMP_MAX)
6304                        ) THEN
6305
6306                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6307
6308                    END;
6309
6310                    IF (
6311                    (TEST.SECONDARY = MT_BLOCK_SIZE)
6312                    AND
6313                    (TEST.NUMBER > 0)
6314                    AND
6315                    (TEST.NUMBER < 20)
6316                    ) THEN
6317
6318                        LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6319
6320                END;    { IF NUMEER_VALUED }
6321
6322                IF (
6323                (SEC_TYPE[TEST.SECONDARY].STR)
6324                OR
6325                (TEST.PRIMARY = TITLE)
6326                ) THEN
6327
6328                BEGIN
6329
6330                    IF TEST.PRIMARY = DATE THEN
6331
6332                        WRITE ('Date-str)',ANSI_REVERSE,'[-]',ANSI_RESET,'      : ')
6333
6334                    ELSE IF TEST.SECONDARY = NAME$ THEN
6335
6336                        WRITE ('1-32 chars)[null]       : ')
6337
6338                    ELSE
6339
6340                        WRITE ('1-126 chars)[null]',CRLF_SHIFT,': ');
```

```
6341
6342                    IF EOF (INPUT) THEN
6343
6344                    BEGIN
6345
6346                        RESET (INPUT);
6347                        LIB$SIGNAL (EDF$_CTRLZ,0,0,0);
6348
6349                    END;
6350
6351                    READLN (TEMP_STRING255);
6352                    WRITELN (CRLF);
6353
6354                    TEST.STRING := NULL_STRING;
6355                    STR$TRIM (TEST.STRING,TEMP_STRING255);
6356                    LIB$SCOPY_DXDX (TEST.STRING,INPUT_DESC);
6357                    PARAM_BLOCK.TPA$L_TOKENPTR  := INPUT_DESC.DSC$A_POINTER::UNSIGNED;
6358                    PARAM_BLOCK.TPA$L_TOKENCNT  := INPUT_DESC.DSC$W_LENGTH;
6359
6360                    { +
6361                    If we're journaling our input, save a copy of it to the
6362                    journal file.
6363                    - }
6364                    IF JOURNAL_ENABLED THEN
6365
6366                        IF TEST.STRING.DSC$W_LENGTH > 0 THEN
6367
6368                            WRITELN (
6369                                JOURNAL_FILE,
6370                                TEST.STRING.DSC$A_POINTER^:
6371                                    TEST.STRING.DSC$W_LENGTH
6372                                )
6373
6374                        ELSE
6375
6376                            WRITELN (JOURNAL_FILE);
6377
6378                    IF TEST.PRIMARY = DATE THEN
6379
6380                    BEGIN
6381
6382                        STR$UPCASE (TEST.STRING,TEST.STRING);
6383
6384                        IF TEST.STRING.DSC$W_LENGTH = 0 THEN
6385
6386                        BEGIN
6387
6388                            STR$FREE1_DX (TEST.STRING);
6389                            LIB$SIGNAL (EDF$_NODEFAULT,0,0,0);
6390
6391                        END;
6392
6393                        { +
6394                        Copy the upcased string back into the temp_string255
6395                        for the test.
6396                        - }
6397                        FOR I := 1 TO TEST.STRING.DSC$W_LENGTH DO
```

```
6398
6399                        TEMP_STRING255[I]   := TEST.STRING.DSC$A_POINTER^[I];
6400
6401                    IF (
6402                    NOT ODD ( $BINTIM (TEMP_STRING255,QUAD_TIME) )
6403                    ) THEN
6404
6405                    BEGIN
6406
6407                        STR$FREE1_DX (TEST.STRING);
6408                        LIB$SIGNAL (EDF$_BADSYNTAX,0,0,0);
6409
6410                    END;
6411
6412                END;        { IF TEST.PRIMARY = DATE }
6413
6414                IF (
6415                (
6416                (TEST.SECONDARY = NAME$)
6417                AND
6418                (TEST.STRING.DSC$W_LENGTH > 32)
6419                )
6420                OR
6421                (TEST.STRING.DSC$W_LENGTH > 126)
6422                ) THEN
6423
6424                BEGIN
6425
6426                    STR$FREE1_DX (TEST.STRING);
6427                    LIB$SIGNAL (EDF$_BADSYNTAX,0,0,0);
6428
6429                    END;
6430
6431                END;    { IF STRING_VALUED }
6432
6433                IF SEC_TYPE[TEST.SECONDARY].SW THEN
6434
6435                BEGIN
6436
6437                    WRITE ('Yes/No)',ANSI_REVERSE,'[-]',ANSI_RESET,'    : ');
6438                    PARSE_INPUT (
6439                                IADDRESS (EDF$AB_YES_NO_TABLE_KEY),
6440                                IADDRESS (EDF$AB_YES_NO_TABLE_STA),
6441                                FALSE,
6442                                0
6443                                );
6444
6445                    TEST.SWITCH := (INPUT_VALUE = EDF$K_YES);
6446
6447                    IF (
6448                    (TEST.PRIMARY = KEY)
6449                    AND
6450                    (TEST.PRINUM = 0)
6451                    AND
6452                    (TEST.SECONDARY = CHANGES)
6453                    AND
6454                    (TEST.SWITCH = TRUE)
```

```
6455                ) THEN
6456
6457                    LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6458
6459                IF (
6460                (TEST.PRIMARY = RECORD$)
6461                AND
6462                (TEST.SECONDARY = BLOCK_SPAN)
6463                AND
6464                (TEST.SWITCH = TRUE)
6465                ) THEN
6466
6467                BEGIN
6468
6469                    IF FIND_OBJECT (SEC,FILE$,0,ORGANIZATION,0) THEN
6470
6471                    BEGIN
6472
6473                        IF DEF_CURRENT^.QUALIFIER <> FDL$C_SEQ THEN
6474
6475                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6476
6477                    END;
6478
6479                END;
6480            END;    { IF SWITCH_VALUED }
6481
6482            IF TEST.SECONDARY = OWNER THEN
6483
6484            BEGIN
6485
6486                EDF$GL_OWNER_UIC    := 0;
6487
6488                WRITE ('UIC-str)',ANSI_REVERSE,'[-]',ANSI_RESET,'    : ');
6489                PARSE_INPUT (
6490                            IADDRESS (EDF$AB_UIC_TABLE_KEY),
6491                            IADDRESS (EDF$AB_UIC_TABLE_STA),
6492                            FALSE,
6493                            0
6494                            );
6495
6496                TEST.OWNER_UIC      := EDF$GL_OWNER_UIC;
6497
6498            END;    { IF TEST.SECONDARY = OWNER }
6499
6500            IF TEST.SECONDARY = PROTECTION THEN
6501
6502            BEGIN
6503
6504                FOR I := 0 TO 31 DO
6505
6506                    EDF$GL_PROT_MASK[I]     := FALSE;
6507
6508                WRITE ('Prot-str)',ANSI_REVERSE,'[-]',ANSI_RESET,CRLF_SHIFT,': ');
6509                PARSE_INPUT (
6510                            IADDRESS (EDF$AB_PROT_TABLE_KEY),
6511
```

```
6512                        IADDRESS (EDF$AB_PROT_TABLE_STA),
6513                        FALSE,
6514                        0
6515                        );
6516
6517            TEST.PROT_MASK      := EDF$GL_PROT_MASK;
6518
6519        END;    { IF TEST.SECONDARY = PROTECTION }
6520
6521        IF TEST.SECONDARY = POSITION$ THEN
6522
6523        BEGIN
6524
6525            WRITE ('Keyword)',ANSI_REVERSE,'[-]',ANSI_RESET,'    : ');
6526            PARSE_INPUT (
6527                        IADDRESS (EDF$AB_POSIT_TABLE_KEY),
6528                        IADDRESS (EDF$AB_POSIT_TABLE_STA),
6529                        FALSE,
6530                        0
6531                        );
6532
6533            TEST.QUALIFIER      := INPUT_VALUE;
6534
6535            IF NOT (TEST.QUALIFIER IN [ FDL$C_NOPOS, FDL$C_ANYPOS ]) THEN
6536
6537            BEGIN
6538
6539                WRITE (CRLF_SHIFT,'Enter POSITION value         (');
6540
6541                CASE TEST.QUALIFIER OF
6542
6543                    FDL$C_CLUSPOS, FDL$C_CYLPOS, FDL$C_LOGPOS, FDL$C_VIRPOS :
6544
6545                    BEGIN
6546
6547                        WRITE ('0-1Giga)',ANSI_REVERSE,'[-]',ANSI_RESET,'       : ');
6548                        NUMBER_INPUT (TEST.NUMBER,FALSE,0);
6549
6550                    END;
6551
6552                    FDL$C_FIDPOS :
6553
6554                    BEGIN
6555
6556                        EDF$GL_FID1     := 0;
6557                        EDF$GL_FID2     := 0;
6558                        EDF$GL_FID3     := 0;
6559
6560                        WRITE ('FID-str)',ANSI_REVERSE,'[-]',ANSI_RESET,'       : ');
6561                        PARSE_INPUT (
6562                                    IADDRESS (EDF$AB_FID_TABLE_KEY),
6563                                    IADDRESS (EDF$AB_FID_TABLE_STA),
6564                                    FALSE,
6565                                    0
6566                                    );
6567
6568                        TEST.FID1               := EDF$GL_FID1;
```

```
6569                        TEST.FID2               := EDF$GL_FID2;
6570                        TEST.FID3               := EDF$GL_FID3;
6571
6572                    END;
6573
6574                    FDL$C_FNMPOS :
6575
6576                    BEGIN
6577
6578                        WRITE ('1-109 chars)[  .L]',CRLF_SHIFT,': ');
6579
6580                        IF EOF (INPUT) THEN
6581
6582                        BEGIN
6583
6584                            RESET (INPUT);
6585                            LIB$SIGNAL (EDF$_CTRLZ,0,0,0);
6586
6587                        END;
6588
6589                        READLN (TEMP_STRING255);
6590                        WRITELN (CRLF);
6591
6592                        TEST.STRING     := NULL_STRING;
6593                        STR$TRIM (TEST.STRING,TEMP_STRING255);
6594                        LIB$SCOPY_DXDX (TEST.STRING,INPUT_DESC);
6595                        PARAM_BLOCK.TPA$L_TOKENPTR      := INPUT_DESC.DSC$A_POINTER::UNSIGNED;
6596                        PARAM_BLOCK.TPA$L_TOKENCNT      := INPUT_DESC.DSC$W_LENGTH;
6597
6598                        { +
6599                        If we're journaling our input, save a copy of it to the
6600                        journal file.
6601                        - }
6602                        IF JOURNAL_ENABLED THEN
6603
6604                            IF TEST.STRING.DSC$W_LENGTH > 0 THEN
6605
6606                                WRITELN (
6607                                    JOURNAL_FILE,
6608                                    TEST.STRING.DSC$A_POINTER^:
6609                                        TEST.STRING.DSC$W_LENGTH
6610                                    )
6611
6612                            ELSE
6613
6614                                WRITELN (JOURNAL_FILE);
6615
6616                        IF TEST.STRING.DSC$W_LENGTH > 109 THEN
6617
6618                            LIB$SIGNAL (EDF$_BADSYNTAX,0,0,0);
6619
6620                    END;
6621
6622                OTHERWISE
6623
6624                    { NULL-STATEMENT } ;
6625
```

```
6626                        END;    { CASE }
6627
6628                    END;        { IF NOT (TEST.QUALIFIER IN [ FDL$C_NOPOS, FDL$C_ANYPOS ]) }
6629
6630                END;    { IF TEST.SECONDARY = POSITION$ }
6631
6632                IF (
6633                (TEST.SECONDARY = NULL_VALUE)
6634                OR
6635                (TEST.SECONDARY = MT_PROTECTION)
6636                ) THEN
6637
6638                BEGIN
6639
6640                    WRITE ('''char''/num)',ANSI_REVERSE,'[-]',ANSI_RESET,'      : ');
6641
6642
6643                    IF EOF (INPUT) THEN
6644
6645                    BEGIN
6646
6647                        RESET (INPUT);
6648                        LIB$SIGNAL (EDF$_CTRLZ,0,0,0);
6649
6650                    END;
6651
6652                    READLN (TEMP_STRING255);
6653                    WRITELN (CRLF);
6654
6655                    TEMP_DESC   := NULL_STRING;
6656                    STR$TRIM (TEMP_DESC,TEMP_STRING255);
6657                    LIB$SCOPY_DXDX (TEMP_DESC,INPUT_DESC);
6658                    PARAM_BLOCK.TPA$L_TOKENPTR  := INPUT_DESC.DSC$A_POINTER::UNSIGNED;
6659                    PARAM_BLOCK.TPA$L_TOKENCNT  := INPUT_DESC.DSC$W_LENGTH;
6660
6661                    { +
6662                    If we're journaling our input, save a copy of it to the
6663                    journal file.
6664                    - }
6665                    IF JOURNAL_ENABLED THEN
6666
6667                        IF TEMP_DESC.DSC$W_LENGTH > 0 THEN
6668
6669                            WRITELN (
6670                                JOURNAL_FILE,
6671                                TEMP_DESC.DSC$A_POINTER^:
6672                                    TEMP_DESC.DSC$W_LENGTH
6673                                )
6674
6675                        ELSE
6676
6677                            WRITELN (JOURNAL_FILE);
6678
6679                    IF TEMP_DESC.DSC$W_LENGTH = 0 THEN
6680
6681                    BEGIN
6682
```

```
6683                        STR$FREE1_DX (TEMP_DESC);
6684                        LIB$SIGNAL (EDF$_NODEFAULT,0,0,0);
6685
6686                    END;
6687
6688                    ISTATUS             := OTS$CVT_TI_L (TEMP_DESC,TEMP_INT);
6689
6690                    IF ODD (ISTATUS) THEN
6691
6692                        TEST.NUMBER     := TEMP_INT
6693
6694                    ELSE IF (
6695                    (TEMP_DESC.DSC$A_POINTER^[1] <> APOSTROPHE)
6696                    OR
6697                    (TEMP_DESC.DSC$A_POINTER^[3] <> APOSTROPHE)
6698                    ) THEN
6699
6700                    BEGIN
6701
6702                        STR$FREE1_DX (TEMP_DESC);
6703                        LIB$SIGNAL (EDF$_BADSYNTAX,0,0,0);
6704
6705                    END
6706
6707                    ELSE
6708
6709                        TEST.NUMBER     := ORD (TEMP_DESC.DSC$A_POINTER^[2]);
6710
6711                    IF TEST.SECONDARY = MT_PROTECTION THEN
6712
6713                    BEGIN
6714
6715                        { +
6716                        Make sure it's a legal ANSI-a character.
6717                        - }
6718                        IF (
6719                        (TEST.NUMBER < %X20)    { SPACE }
6720                        OR
6721                        (TEST.NUMBER > %X5A)    { CAPITAL Z }
6722                        OR
6723                        (TEST.NUMBER = %X23)    { # }
6724                        OR
6725                        (TEST.NUMBER = %X24)    { $ }
6726                        OR
6727                        (TEST.NUMBER = %X40)    { @ }
6728                        ) THEN
6729
6730                        BEGIN
6731
6732                            STR$FREE1_DX (TEMP_DESC);
6733                            LIB$SIGNAL (EDF$_BADVALUE,0,0,0);
6734
6735                        END;
6736
6737                    END; { IF TEST.SECONDARY = MT_PROTECTION }
6738
6739                END;    { IF TEST.SECONDARY = NULL_VALUE OR MT_PROTECTION }
```

```
6740
6741                    { +
6742                    Make sure this is true only for one cycle.
6743                    - }
6744                    TEMP_FULL_PROMPT        := FALSE;
6745
6746                END;        { THE_QUESTION }
6747
6748            BEGIN
6749
6750                IF TEST.PRIMARY = TITLE THEN
6751
6752                    TEST.OBJECT_TYPE        := PRI
6753
6754                ELSE
6755
6756                    TEST.OBJECT_TYPE        := SEC;
6757
6758                { +
6759                Keep at it until the user gets it right.
6760                - }
6761                REPEAT
6762
6763                    THE_QUESTION;
6764
6765                UNTIL NOT SYS$INPUT_ERROR;
6766
6767                STR$FREE1_DX (INPUT_DESC);
6768
6769            END;            { ASK_TEST_SECONDARY_VALUE }
6770
6771            END.
6772                { End of file: SRC$:EDFASK.PAS }
```

```
                                .TITLE  EDFASK
                                .IDENT  \V04-000\

00000                           .PSECT  $CODE,PIC,CON,REL,LCL,SHR,EXE,RD,NOWRT,2

74 65 6C 20 32 20 65 68 74 20 65 70 79 54   00000 C.AAA:   .ASCII  \Type the 2 letter mnemonic of the select\-
6F 20 63 69 6E 6F 6D 65 6E 6D 20 72 65 74   0000E                  \ed option.\<0><0>
64 65 74 63 65 6C 65 73 20 65 68 74 20 66   0001C
                  00 00 2E 6E 6F 69 74 70   0002A
6F 20 74 73 20 65 68 74 20 73 69 20 73 69   00034 C.AAB:   .ASCII  \This is the starting byte of the key or \-
68 54 66 6F 20 65 74 79 62 20 67 6E 69 74   00042                  \key segment.\
72 61 65 6B 20 72 6F 20 79 65 6B 20 65 68   00050
                  74 20 2E 74 6E 65 6D 67   0005E
65 73 20 74 6F 20 73 72 65 66 65 72 20 73   00068 C.AAC:   .ASCII  \This refers to records that are added to\-
69 68 54 20 74 61 68 74 20 73 64 72 6F 63   00076                  \ the\
65 72 20 74 20 6F 74 20 64 65 64 64 61 20   00084
                                  65 72 61   00092
65 68 20 74 69 20 72 65 74 66 61 20 65 6C   00094 C.AAD:   .ASCII  \file after it is initially loaded.\<0><0>
69 66 6C 20 79 6C 6C 61 69 74 69 6E 69 20   000A2
                  73 69 00 00 2E 64 65 64   000B0
61 6F 6F 20 73 65 6C 62 61 6E 65 20 73 69   00098 C.AAE:   .ASCII  \This enables or disables the RMS option.\
68 54 65 68 74 20 73 65 6C 62 61 73 69 64   000C6
20 72 2E 6E 6F 69 74 70 6F 20 53 4D 52 20   000D4
6F 20 73 65 6C 62 61 6E 65 20 73 69 68 54   000E0 C.AAF:   .ASCII  \This enables or disables the RMS option.\
65 68 74 20 73 65 6C 62 61 73 69 64 20 72   000EE
2E 6E 6F 69 74 70 6F 20 53 4D 52 20         000FC
65 6B 20 67 6E 69 72 74 73 20 68 63 61 45   00108 C.AAG:   .ASCII  \Each string key may consist of up to 8 p\-
20 74 73 69 73 6E 6F 63 20 79 61 6D 20 79   00116                  \arts.\<0><0><0>
72 61 70 20 38 20 6F 74 20 70 75 20 66 6F   00124
                  00 00 00 2E 73 74 61 74   00132
20 79 6C 6C 61 75 73 75 20 65 73 65 68 54   00138 C.AAH:   .ASCII  \These usually increase the speed of file\-
73 20 65 68 74 20 65 73 61 65 72 63 6E 69   00146                  \ sharing,\<0><0><0>
73 20 65 6C 69 66 20 66 6F 20 64 65 65 70   00154
                  00 00 00 2C 67 6E 69 72   00162
61 68 65 73 6E 65 70 78 65 20 65 68 74 61   0016C C.AAI:   .ASCII  \at the expense of using more physical me\-
65 72 6F 6D 20 67 6E 69 73 75 20 66 6F 20   0017A                  \mory.\<0><0><0>
6F 6D 65 6D 20 6C 61 63 69 73 79 68 70 20   00188
                  00 00 00 2E 79 72 61 74   00196
6E 61 65 6D 20 63 69 74 61 6D 6F 74 75 41   0019C C.AAJ:   .ASCII  \Automatic means the default answers will\-
20 74 6C 75 61 66 65 64 20 65 68 74 20 73   001AA                  \ be used without\
62 20 6C 6C 69 77 20 73 72 65 77 73 6E 61   001B8
74 75 6F 68 74 69 77 20 64 65 73 75 20 65   001C6
6F 63 20 72 6F 66 20 67 6E 69 74 69 61 77   001D4 C.AAK:   .ASCII  \waiting for confirmation.\<0><0><0>
00 00 00 2E 6E 6F 69 74 61 6D 72 69 66 6E   001E2
20 73 6C 6F 72 74 6E 6F 63 20 73 69 68 54   001F0 C.AAL:   .ASCII  \This controls whether full menus are dis\-
6D 20 6C 6C 75 66 20 72 65 68 74 65 68 77   001FE                  \played.\<0>
6C 70 73 69 64 20 65 72 61 20 73 75 6E 65   0020C
                  00 2E 64 65 79 61 65 74   0021A
65 74 65 64 20 6C 6C 69 77 20 73 69 68 54   00220 C.AAM:   .ASCII  \This will determine the allocation of th\-
6F 6C 6C 61 20 65 68 74 20 65 6E 69 6D 72   0022E                  \e file.\<0>
20 65 68 74 20 66 6F 20 6E 6F 69 74 61 63   0023C
                  00 2E 65 6C 69 66 74 20   0024A
6F 74 20 73 72 65 66 65 72 20 73 69 68 54   00250 C.AAN:   .ASCII  \This refers to the order of the initial \-
20 66 6F 20 72 65 64 72 6F 20 65 68 74 20   0025E                  \records loaded.\<0>
65 72 20 6C 61 69 74 69 6E 69 20 65 68 74   0026C
00 2E 64 65 64 61 6F 6C 20 73 64 72 6F 63   0027A
20 65 68 74 20 65 72 61 20 65 73 65 68 54   00288 C.AAO:   .ASCII  \These are the records initially loaded i\-
```

```
61 69 74 69 6E 69 20 73 64 72 6F 63 65 72   00296                   \nto the file.\<0><0><0>
74 6E 69 20 64 65 64 61 6F 6C 20 79 6C 6C   002A4
00 00 00 2E 65 6C 69 66 20 65 68 74 20 6F   002B2
69 77 20 65 6C 69 66 20 65 68 74 20 66 49   002C0 C.AAP:  .ASCII  \If the file will have no ''Load'' operatio\-
6F 4C 22 20 6F 6E 20 65 76 61 68 20 6C 6C   002CE                 \n, specify ''0''.\<0>
2C 6E 6F 69 74 61 72 65 70 6F 20 22 64 61   002DC
00 2E 22 30 22 20 79 66 69 63 65 70 73 20   002EA
20 65 68 74 20 65 72 61 20 65 73 65 68 54   002F8 C.AAQ:  .ASCII  \These are the records added after the in\-
20 64 65 64 64 61 20 73 64 72 6F 63 65 72   00306                 \itial file load.\
74 69 6E 69 20 65 68 74 20 72 65 74 66 61   00314
2E 64 61 6F 6C 20 65 6C 69 66 20 6C 61 69   00322
65 72 20 68 63 61 65 20 2C 6F 6E 20 66 49   00330 C.AAR:  .ASCII  \If no, each record plus overhead must fi\-
72 65 76 6F 20 73 75 6C 70 20 64 72 6F 63   0033E                 \t in a disk block.\<0><0>
20 74 69 66 20 74 73 75 6D 20 64 61 65 68   0034C
63 6F 6C 62 20 6B 73 69 64 20 61 20 6E 69   0035A
                              00 00 2E 6B   00368
61 70 73 20 65 6D 6F 73 20 2C 6F 73 6C 41   0036C C.AAS:  .ASCII  \Also, some space may be wasted at the en\-
74 73 61 77 20 65 62 20 79 61 6D 20 65 63   0037A                 \d of blocks.\
20 64 6E 65 20 65 68 74 20 74 61 20 64 65   00388
                  2E 73 6B 63 6F 6C 62 20   00396
65 6C 20 65 68 74 20 73 69 20 73 69 68 54   003A0 C.AAT:  .ASCII  \This is the length of the key (segment) \-
65 6B 20 65 68 74 20 66 6F 20 68 74 67 6E   003AE                 \in bytes.\<0><0><0>
6E 69 20 29 74 6E 65 6D 67 65 73 28 20 79   003BC
                  00 00 00 2E 73 65 74 79   003CA
65 73 2D 69 74 6C 75 6D 20 68 74 69 57 28   003D4 C.AAU:  .ASCII  \(With multi-segment keys, answer ''0'' aft\-
6E 61 20 2C 73 79 65 6B 20 74 6E 65 6D 67   003E2                 \er the last segment.)\<0><0><0>
72 65 74 66 61 20 22 30 22 20 72 65 77 73   003F0
6D 67 65 73 20 74 73 61 6C 20 65 68 74 20   003FE
                  00 00 00 29 2E 74 6E 65   0040C
6F 74 20 73 72 65 66 65 72 20 73 69 68 54   00414 C.AAV:  .ASCII  \This refers to the structure level of th\-
65 72 75 74 63 75 72 74 73 20 65 68 74 20   00422                 \e data file.\
20 65 68 74 20 66 6F 20 6C 65 76 65 6C 20   00430
                  2E 65 6C 69 66 20 61 74   0043E
6C 20 30 20 66 6F 20 65 75 6C 61 76 20 41   00448 C.AAW:  .ASCII  \A value of 0 lets RMS choose an appropri\-
65 73 6C 6F 68 63 20 53 4D 52 20 73 74 65   00456                 \ate prolog.\<0>
74 61 69 72 70 6F 72 70 70 61 20 6E 61 20   00464
                  00 2E 67 6F 6C 6F 72 70   00472
2F 65 7A 79 6C 61 6E 41 20 6E 61 20 66 49   0047C C.AAX:  .ASCII  \If an Analyze/RMS indicates little compr\-
20 73 65 74 61 63 69 64 6E 69 20 53 4D 52   0048A                 \ession is acheived\<0><0>
73 65 72 70 6D 6F 63 20 65 6C 74 74 69 6C   00498
76 69 65 68 63 61 20 73 69 20 6E 6F 69 73   004A6
                              00 00 64 65   004B4
6F 4E 20 72 65 77 73 6E 61 20 6E 65 68 74   004B8 C.AAY:  .ASCII  \then answer No, otherwise it is usually \-
74 69 20 65 73 69 77 72 65 68 74 6F 20 2C   004C6                 \better to answer Yes.\<0><0><0>
65 62 20 79 6C 6C 61 75 73 75 20 73 69 20   004D4
72 65 77 73 6E 61 20 6F 74 20 72 65 74 74   004E2
                  00 00 00 2E 73 65 59 20   004F0
55 46 2F 45 43 49 56 45 44 20 57 4F 48 53   004F8 C.AAZ:  .ASCII  \SHOW DEVICE/FULL can be used to determin\-
64 65 73 75 20 65 62 20 6E 61 63 20 4C 4C   00506                 \e this value.\<0><0><0>
20 65 6E 69 6D 72 65 74 65 64 20 6F 74 20   00514
00 00 00 2E 65 75 6C 61 76 20 73 69 68 74   00522
6F 74 20 73 72 65 66 65 72 20 73 69 68 54   00530 C.ABA:  .ASCII  \This refers to the orderring of addition\-
67 6E 69 72 72 65 64 72 6F 20 65 68 74 20   0053E                 \al records.\<0>
6C 61 6E 6F 69 74 69 64 61 20 66 6F 20 74   0054C
                  00 2E 73 64 72 6F 63 65   0055A
73 69 20 65 67 6E 61 72 20 6C 61 67 65 4C   00564 C.ABB:  .ASCII  \Legal range is 1 to 63 blocks per bucket\-
6B 63 6F 6C 62 20 33 36 20 6F 74 20 31 20   00572                 \, and buckets must\<0><0>
```

```
                                                                00580
                                                                0058E
                                                                0059C
                                                                005A0 C.ABC:  .ASCII  \be large enough to hold at least 1 recor\-
                                                                005AE                 \d plus overhead.\
                                                                005BC
                                                                005CA
                                                                005D8 C.ABD:  .ASCII  \Smaller_Buffers: less memory and RMS pro\-
                                                                005E6                 \cessing used\
                                                                005F4
                                                                00602
                                                                0060C C.ABE:  .ASCII  \Flatter_Files:   fewer actual disk acces\-
                                                                0061A                 \ses needed\<0><0>
                                                                00628
                                                                00636
                                                                00640 C.ABF:  .ASCII  \Legal values: Fast_Convert, NoFast_Conve\-
                                                                0064E                 \rt, RMS_Puts\
                                                                0065C
                                                                0066A
                                                                00674 C.ABG:  .ASCII  \Fast_Convert:   using the VAX-11 Convert\-
                                                                00682                 \/Fast_Load option\<0><0><0>
                                                                00690
                                                                0069E
                                                                006AC
                                                                006B0 C.ABH:  .ASCII  \NoFast_Convert: using the VAX-11 Convert\-
                                                                006BE                 \/NoFast_Load option\<0>
                                                                006CC
                                                                006DA
                                                                006E8
                                                                006EC C.ABI:  .ASCII  \RMS_Puts:       writing to a file from a\-
                                                                006FA                 \ High Level Language\
                                                                00708
                                                                00716
                                                                00724
                                                                00728 C.ABJ:  .ASCII  \This is the initial file loading fill fa\-
                                                                00736                 \ctor.\<0><0><0>
                                                                00744
                                                                00752
                                                                00758 C.ABK:  .ASCII  \Legal types: Bin2 Bin4 Bin8 Int2 Int4 In\-
                                                                00766                 \t8 Decimal String\<0><0><0>
                                                                00774
                                                                00782
                                                                00790
                                                                00794 C.ABL:  .ASCII  \Use\<0>
                                                                00798 C.ABM:  .ASCII  \"Binx" types for unsigned binary keys of\-
                                                                007A6                 \ 2, 4 or 8 bytes,\<0><0><0>
                                                                007B4
                                                                007C2
                                                                007D0
                                                                007D4 C.ABN:  .ASCII  \"Intx" types for signed binary key of 2,\-
                                                                007E2                 \ 4 or 8 bytes,\<0><0>
                                                                007F0
                                                                007FE
                                                                0080C C.ABO:  .ASCII  \"Decimal" type for packed decimal key of\-
                                                                0081A                 \ 1 to 16 bytes,\<0>
                                                                00828
                                                                00836
```

```
                                                
```

```
20 64 65 64 64 41 20 66 6F 20 72 65 62 6D    00AC6                   <0><0>
                  00 00 00 28 73 63 65 52    00AD4
                  5B 29 61 67 69 47 31 2D    00ADC C.ACG:  .ASCII  \-1Giga)[\
                              00 20 3A 09    00AE4 C.ACH:  .ASCII  <9>\: \<0>
                              00 20 3A 20    00AE8 C.ACI:  .ASCII  \ : \<0>
79 65 4B 20 3A 64 6E 75 6F 62 20 77 6F 4C    00AEC C.ACJ:  .ASCII  \Low bound: Key\<0><0>
                                 00 00       00AFA
      00 2D 31 28 09 68 74 67 6E 65 4C 20    00AFC C.ACK:  .ASCII  \ Length\<9>\(1-\<0>
               00 20 3A 09 5D 31 5B 29       00B08 C.ACL:  .ASCII  \)[1]\<9>\: \<0>
65 4B 20 3A 64 6E 75 6F 62 20 68 67 69 48    00B10 C.ACM:  .ASCII  \High bound: Key\<0>
                                 00 79       00B1E
         00 28 09 68 74 67 6E 65 4C 20       00B20 C.ACN:  .ASCII  \ Length\<9>\(\<0>
                              5B 29          00B2A C.ACO:  .ASCII  \)[\
                        20 3A 09 5D          00B2C C.ACP:  .ASCII  \]\<9>\: \
63 65 52 20 3A 64 6E 75 6F 62 20 77 6F 4C    00B30 C.ACQ:  .ASCII  \Low bound: Record Size\<9><9>\(1-\<0>
00 2D 31 28 09 09 65 7A 69 53 20 64 72 6F    00B3E
               00 20 3A 09 5D 31 5B 29       00B4C C.ACR:  .ASCII  \)[1]\<9>\: \<0>
65 52 20 3A 64 6E 75 6F 62 20 68 67 69 48    00B54 C.ACS:  .ASCII  \High bound: Record Size\<9><9>\(\<0><0>
00 00 28 09 09 65 7A 69 53 20 64 72 6F 63    00B62
               00 5D 30 30 30 31 5B 29       00B70 C.ACT:  .ASCII  \)[1000]\<0>
                              00 20 3A 09    00B78 C.ACU:  .ASCII  <9>\: \<0>
                              00 20 3A 20    00B7C C.ACV:  .ASCII  \ : \<0>
79 65 4B 20 3A 64 6E 75 6F 62 20 77 6F 4C    00B80 C.ACW:  .ASCII  \Low bound: Key\<0><0>
                                 00 00       00B8E
28 09 25 20 6C 6C 69 46 20 74 69 6E 49 20    00B90 C.ACX:  .ASCII  \ Init Fill %\<9>\(50-100)[50]\<9>\: \
20 3A 09 5D 30 35 5B 29 30 30 31 2D 30 35    00B9E
65 4B 20 3A 64 6E 75 6F 62 20 68 67 69 48    00BAC C.ACY:  .ASCII  \High bound: Key\<0>
                                 00 79       00BBA
28 09 25 20 6C 6C 69 46 20 74 69 6E 49 20    00BBC C.ACZ:  .ASCII  \ Init Fill %\<9>\(50-100)[100]\<9>\: \<0>
3A 09 5D 30 30 31 5B 29 30 30 31 2D 30 35    00BCA
                              00 20 09 09    00BD8
                                             00BDA C.ADA:  .ASCII  <9><9>
20 65 6C 74 69 54 20 74 70 69 72 63 53 20    00BDC C.ADB:  .ASCII  \ Script Title Selection \
               20 6E 6F 69 74 63 65 6C 65 53 00BEA
6C 65 64 6F 6D 09 09 79 65 4B 5F 64 64 41    00BF4 C.ADC:  .ASCII  \Add_Key\<9><9>\modeling and addition of \-
69 74 69 64 64 61 20 64 6E 61 20 67 6E 69    00C02                 \a new index's parameters\<0><0>
6E 69 20 77 65 6E 20 61 20 66 6F 20 6E 6F    00C10
65 74 65 6D 61 72 61 70 20 73 27 78 65 64    00C1E
                              00 00 73 72    00C2C
6D 65 72 09 79 65 4B 5F 65 74 65 6C 65 44    00C30 C.ADD:  .ASCII  \Delete_Key\<9>\removal of the highest in\-
69 68 20 65 68 74 20 66 6F 20 6C 61 76 6F    00C3E                 \dex's parameters\
20 73 27 78 65 64 6E 69 20 74 73 65 68 67    00C4C
               73 72 65 74 65 6D 61 72 61 70 00C5A                                                    75
6C 65 64 6F 6D 09 09 64 65 78 65 64 6E 49    00C64 C.ADE:  .ASCII  \Indexed\<9><9>\modeling of parameters fo\-   68
74 65 6D 61 72 61 70 20 66 6F 20 67 6E 69    00C72                 \r an entire Indexed file\<0><0>           61
74 6E 65 20 6E 61 20 72 6F 66 20 73 72 65    00C80                                                      00
69 66 20 64 65 78 65 64 6E 49 20 65 72 69    00C8E
                              00 00 65 6C    00C9C
6E 69 6E 75 74 09 65 7A 69 6D 69 74 70 4F    00CA0 C.ADF:  .ASCII  \Optimize\<9>\tuning of all indices' para\-
63 69 64 6E 69 20 6C 6C 61 20 66 6F 20 67    00CAE                 \meters using file statistics\
73 72 65 74 65 6D 61 72 61 70 20 27 73 65    00CBC
74 73 20 65 6C 69 66 20 67 6E 69 73 75 20    00CCA
               73 63 69 74 73 69 74 61 6C    00CD8
63 65 6C 65 73 09 65 76 69 74 61 6C 65 52    00CE0 C.ADG:  .ASCII  \Relative\<9>\selection of parameters for\-
65 6D 61 72 61 70 20 66 6F 20 6E 6F 69 74    00CEE                 \ a Relative file\                         00
6C 65 52 20 61 20 72 6F 66 20 73 72 65 74    00CFC                                                            2F
               65 6C 69 66 20 65 76 69 74 61 00D0A
```

```
                                                00D14 C.ADH:  .ASCII  \Sequential\<9>\selection of parameters f\-
6C 65 73 09 6C 61 69 74 6E 65 75 71 65 53       00D22                 \or a Sequential file\
61 72 61 70 20 66 6F 20 6E 6F 69 74 63 65       00D30
53 20 61 20 72 6F 66 20 73 72 65 74 65 6D       00D3E
65 6C 69 66 20 6C 61 69 74 6E 65 75 71 65       00D4C C.ADI:  .ASCII  \Touchup\<9><9>\remodeling of parameters \-
64 6F 6D 65 72 09 09 70 75 68 63 75 6F 54       00D5A                 \for a particular index\
6D 61 72 61 70 20 66 6F 20 67 6E 69 6C 65       00D68
61 70 20 61 20 72 6F 66 20 73 72 65 74 65       00D76
78 65 64 6E 69 20 72 61 6C 75 63 69 74 72       00D84 C.ADJ:  .ASCII  \(Add_Key Delete_Key Indexed Optimize\
74 65 6C 65 44 20 79 65 4B 5F 64 64 41 28       00D92
20 64 65 78 65 64 6E 49 20 79 65 4B 5F 65       00DA0
                  65 7A 69 6D 69 74 70 4F
75 71 65 53 20 65 76 69 74 61 6C 65 52 20       00DA8 C.ADK:  .ASCII  \ Relative Sequential Touchup)\<0><0><0>
70 75 68 63 75 6F 54 20 6C 61 69 74 6E 65       00DB6
                  00 00 00 29                   00DC4
74 70 69 72 63 53 20 67 6E 69 74 69 64 45       00DC8 C.ADL:  .ASCII  \Editing Script Title\<9><9>\(Keyword)\<0>
6F 77 79 65 4B 28 09 09 65 6C 74 69 54 20       00DD6
                  00 29 64 72                   00DE4
                  00 5D 2D 5B                   00DE8 C.ADM:  .ASCII  \[-]\<0>
                  00 20 3A 09                   00DEC C.ADN:  .ASCII  <9>\: \<0>
20 6F 74 20 22 44 46 22 20 65 70 79 54 28       00DF0 C.ADO:  .ASCII  \(Type "FD" to Finish Design)\
29 6E 67 69 73 65 44 20 68 73 69 6E 69 46       00DFE
72 61 50 20 65 6C 69 46 20 68 63 69 68 57       00E0C C.ADP:  .ASCII  \Which File Parameter\<9>\(Mnemonic)[refr\-
6E 6F 6D 65 6E 4D 28 09 72 65 74 65 6D 61       00E1A                 \esh]\<9>\: \<0>
3A 09 5D 68 73 65 72 66 65 72 5B 29 63 69       00E28
                              00 20             00E36
                  00 00 09 09                   00E38 C.ADQ:  .ASCII  <9><9><0><0>
20 74 72 65 73 6E 69 20 4F 74 09 64 64 41       00E3C C.ADR:  .ASCII  \Add\<9>\to insert one or more lines into\-
69 6C 20 65 72 6F 6D 20 72 6F 20 65 6E 6F       00E4A                 \ the FDL definition\<0>
46 20 65 68 74 20 6F 74 6E 69 20 73 65 6E       00E58
00 6E 6F 69 74 69 6E 69 66 65 64 20 4C 44       00E66
6F 6D 65 72 20 6F 74 09 65 74 65 6C 65 44       00E74 C.ADS:  .ASCII  \Delete\<9>\to remove one or more lines f\-
65 72 6F 6D 20 72 6F 20 65 6E 6F 20 65 76       00E82                 \rom the FDL definition\<0><0>
68 74 20 6D 6F 72 66 20 73 65 6E 69 6C 20       00E90
69 74 69 6E 69 66 65 64 20 4C 44 46 20 65       00E9E
                  00 00 6E 6F                   00EAC
20 65 76 61 65 6C 20 6F 74 09 74 69 78 45       00EB0 C.ADT:  .ASCII  \Exit\<9>\to leave the FDL Editor after c\-
72 6F 74 69 64 45 20 4C 44 46 20 65 68 74       00EBE                 \reating the FDL file\
6E 69 74 61 65 72 63 20 72 65 74 66 61 20       00ECC
65 6C 69 66 20 4C 44 46 20 65 68 74 20 67       00EDA
6E 69 61 74 62 6F 20 6F 74 09 70 6C 65 48       00EE8 C.ADU:  .ASCII  \Help\<9>\to obtain information about the\-
61 20 6E 6F 69 74 61 6D 72 6F 66 6E 69 20       00EF6                 \ FDL Editor\<0>
45 20 4C 44 46 20 65 68 74 20 74 75 6F 62       00F04
                  00 72 6F 74 69 64             00F12
74 69 6E 69 20 6F 74 09 65 6B 6F 76 6E 49       00F18 C.ADV:  .ASCII  \Invoke\<9>\to initiate a script of relat\-
20 74 70 69 72 63 73 20 61 20 65 74 61 69       00F26                 \ed questions\
65 75 71 20 64 65 74 61 6C 65 72 20 66 6F       00F34
                  73 6E 6F 69 74 73             00F42
6E 61 68 63 20 6F 74 09 79 66 69 64 6F 4D       00F48 C.ADW:  .ASCII  \Modify\<9>\to change existing line(s) in\-
69 6C 20 67 6E 69 74 73 69 78 65 20 65 67       00F56                 \ the FDL definition\<0>
46 20 65 68 74 20 6E 69 20 29 73 28 65 6E       00F64
00 6E 6F 69 74 69 6E 69 66 65 64 20 4C 44       00F72
20 74 72 6F 62 61 20 6F 74 09 74 69 75 51       00F80 C.ADX:  .ASCII  \Quit\<9>\to abort the FDL Editor with no\-
72 6F 74 69 64 45 20 4C 44 46 20 65 68 74       00F8E                 \ FDL file creation\<0><0>
66 20 4C 44 46 20 6F 6E 20 68 74 69 77 20       00F9C
00 00 6E 6F 69 74 61 65 72 63 20 65 6C 69       00FAA
79 66 69 63 65 70 73 20 6F 74 09 74 65 53       00FB8 C.ADY:  .ASCII  \Set\<9>\to specify FDL Editor characteri\-
68 63 20 72 6F 74 69 64 45 20 4C 44 46 20       00FC6                 \stics\<0><0><0>
```

```
                                                                00FD4
                                                                00FE2
00 73 63 69 74 73 69 72 65 74 63 61 72 61                      00FE4 C.ADZ:   .ASCII  \View\<9>\to display the current FDL Defi\-
                                           00 00                00FF2                  \nition\<0><0>
61 6C 70 73 69 64 20 6F 74 09 77 65 69 56                       01000
20 74 6E 65 72 72 75 63 20 65 68 74 20 79                       0100E
6E 6F 69 74 69 6E 69 66 65 44 20 4C 44 46                       01010 C.AEA:   .ASCII  \(Add Delete Exit Help Invoke Modify Quit\-
                                           00 00                0101E                  \ Set View)\<0><0>
78 45 20 65 74 65 6C 65 44 20 64 64 41 28                       0102C
65 6B 6F 76 6E 49 20 70 6C 65 48 20 74 69                       0103A
53 20 74 69 75 51 20 79 66 69 64 6F 4D 20                       01044 C.AEB:   .ASCII  \Main Editor Function\<9><9>\(Keyword)[He\-
                  00 00 29 77 65 69 56 20 74 65                 01052                  \lp]\<9>\: \
75 46 20 72 6F 74 69 64 45 20 6E 69 61 4D                       01060
6F 77 79 65 4B 28 09 09 6E 6F 69 74 63 6E                       0106C C.AEC:   .ASCII  \(Automatic Manual)\<0><0>
         20 3A 09 5D 70 6C 65 48 5B 29 64 72                    0107A
6E 61 4D 20 63 69 74 61 6D 6F 74 75 41 28                       01080 C.AED:   .ASCII  \Default responses in scripts\<9>\(Keywor\-
                                 00 00 29 6C 61 75              0108E                  \d)[Auto]\<9>\: \<0>
6E 6F 70 73 65 72 20 74 6C 75 61 66 65 44                       0109C
73 74 70 69 72 63 73 20 6E 69 20 73 65 73                       010AA
74 75 41 5B 29 64 72 6F 77 79 65 4B 28 09                       010B0 C.AEE:   .ASCII  \(Brief Full)\
                  00 20 3A 09 5D 6F                             010BC C.AEF:   .ASCII  \Prompting level for menus\<9>\(Keyword)[\-
         29 6C 6C 75 46 20 66 65 69 72 42 28                    010CA                  \Full]\<9>\: \
65 76 65 6C 20 67 6E 69 74 70 6D 6F 72 50                       010D8
4B 28 09 73 75 6E 65 6D 20 72 6F 66 20 6C                       010E6
09 5D 6C 6C 75 46 5B 29 64 72 6F 77 79 65                       010E8 C.AEG:   .ASCII  \Key\<0>
                                           20 3A                010EC C.AEH:   .ASCII  \ Position\<9><9><0>
                              00 79 65 4B                       010F8 C.AEI:   .ASCII  \SEG\<0>
         00 09 09 6E 6F 69 74 69 73 6F 50 20                    010FC C.AEJ:   .ASCII  <9>\(0-\
                              00 47 45 53                       01100 C.AEK:   .ASCII  \)[0]\<9>\: \<0>
                              2D 30 28 09                       01108 C.AEL:   .ASCII  \Will Added Records be Distributed Evenly\-
                  00 20 3A 09 5D 30 5B 29                       01116                  \ over the\<0><0><0>
63 65 52 20 64 65 64 64 41 20 6C 6C 69 57                       01124
69 72 74 73 69 44 20 65 62 20 73 64 72 6F                       01132
6F 20 79 6C 6E 65 76 45 20 64 65 74 75 62                       0113C C.AEM:   .ASCII  \Initial\<0>
               00 00 00 65 68 74 20 72 65 76                    01144 C.AEN:   .ASCII  \Reloaded\
                  00 6C 61 69 74 69 6E 49                       0114C C.AEO:   .ASCII  \ Range of Pri Key Values\
                  64 65 64 61 6F 6C 65 52                       0115A
20 69 72 50 20 66 6F 20 65 67 6E 61 52 20                       01164 C.AEP:   .ASCII  \(Yes/No)[No]\<9>\: \<0>
               73 65 75 6C 61 56 20 79 65 4B                    01172
3A 09 5D 6F 4E 5B 29 6F 4E 2F 73 65 59 28                       01174 C.AEQ:   .ASCII  \Key\<0>
                                           00 20                01178 C.AER:   .ASCII  \ Changes allowed\<9><9>\(Yes/No)[Yes]\-
                              00 79 65 4B                       01186                  <9>\: \<0><0>
77 6F 6C 6C 61 20 73 65 67 6E 61 68 43 20                       01194
59 5B 29 6F 4E 2F 73 65 59 28 09 09 64 65                       0119C C.AES:   .ASCII  \Key\<0>
                  00 00 20 3A 09 5D 72 65                       011A0 C.AET:   .ASCII  \ Duplicates allowed\<9>\(Yes/No)[\<0>-
                              00 79 65 4B                       011AE                  <0><0>
6C 61 20 73 65 74 61 63 69 6C 70 75 44 20                       011BC
29 6F 4E 2F 73 65 59 28 09 64 65 77 6F 6C                       011C0 C.AEU:   .ASCII  \No]\<9>\: \<0><0>
                              00 00 00 5B                       011C8 C.AEV:   .ASCII  \Yes]\<9>\: \<0>
                  00 00 20 3A 09 5D 6F 4E                       011D0 C.AEW:   .ASCII  \Key\<0>
                  00 20 3A 09 5D 73 65 59                       011D4 C.AEX:   .ASCII  \ Segmentation desired\<9>\(Yes/No)[No]\-
                              00 79 65 4B                       011E2                  <9>\: \<0><0><0>
20 6E 6F 69 74 61 74 6E 65 6D 67 65 53 20                       011F0
4E 2F 73 65 59 28 09 64 65 72 69 73 65 64                       011FC C.AEY:   .ASCII  \Global Buffers desired\<9><9>\(Yes/No)[N\-
         00 00 00 20 3A 09 5D 6F 4E 5B 29 6F                    0120A                  \o]\<9>\: \<0>
73 72 65 66 66 75 42 20 6C 61 62 6F 6C 47                       01218
73 65 59 28 09 09 64 65 72 69 73 65 64 20                       01224 C.AEZ:   .ASCII  \File Capacity in Records\<9>\(0-1Giga)\-
         00 20 3A 09 5D 6F 4E 5B 29 6F 4E 2F
20 79 74 69 63 61 70 61 43 20 65 6C 69 46
```

```
2D 30 28 09 73 64 72 6F 63 65 52 20 6E 69      01232                       <0><0>
                  00 00 29 61 67 69 47 31      01240
                              00 5D 2D 5B      01248 C.AFA:   .ASCII  \[-]\<0>
                              00 20 3A 09      0124C C.AFB:   .ASCII  <9>\: \<0>
6F 63 65 52 20 66 6F 20 72 65 62 6D 75 4E      01250 C.AFC:   .ASCII  \Number of Records that will be Initially\-
20 6C 6C 69 77 20 74 61 68 74 20 73 64 72      0125E                  \ Loaded\<0>
4C 20 79 6C 6C 61 69 74 69 6E 49 20 65 62      0126C
                        00 64 65 64 61 6F      0127A
6F 63 65 52 20 66 6F 20 72 65 62 6D 75 4E      01280 C.AFD:   .ASCII  \Number of Records that will be Reloaded\-
20 6C 6C 69 77 20 74 61 68 74 20 73 64 72      0128E                  <0>
      00 64 65 64 61 6F 6C 65 52 20 65 62      0129C
00 65 6C 69 46 20 65 68 74 20 6F 74 6E 69      012A8 C.AFE:   .ASCII  \into the File\<0><0><0>
                              00 00 00 00      012B6
   29 61 67 69 47 31 2D 30 28 09 09 09 09      012B8 C.AFF:   .ASCII  <9><9><9>\(0-1Giga)\
                              00 5D 2D 5B      012C4 C.AFG:   .ASCII  \[-]\<0>
                              00 20 3A 09      012C8 C.AFH:   .ASCII  <9>\: \<0>
                              00 20 3A 20      012CC C.AFI:   .ASCII  \ : \<0>
                              00 20 3A 09      012D0 C.AFJ:   .ASCII  <9>\: \<0>
20 74 72 65 76 6E 6F 43 5F 74 73 61 46 28      012D4 C.AFK:   .ASCII  \(Fast_Convert NoFast_Convert RMS_Puts)\-
74 72 65 76 6E 6F 43 5F 74 73 61 46 6F 4E      012E2                  <0><0>
      00 00 29 73 74 75 50 5F 53 4D 52 20      012F0
4C 20 65 6C 69 46 20 6C 61 69 74 69 6E 49      012FC C.AFL:   .ASCII  \Initial File Load Method\<9><0><0><0>
00 00 00 09 64 6F 68 74 65 4D 20 64 61 6F      0130A
67 6E 69 64 61 6F 6C 65 52 20 65 6C 69 46      01318 C.AFM:   .ASCII  \File Reloading Method\<9><9><0>
            00 09 09 64 6F 68 74 65 4D 20      01326
74 73 61 46 5B 29 64 72 6F 77 79 65 4B 28      01330 C.AFN:   .ASCII  \(Keyword)[Fast]\<9>\: \<0><0>
                        00 00 20 3A 09 5D      0133E
52 20 6C 61 69 74 69 6E 49 20 6C 6C 69 57      01344 C.AFO:   .ASCII  \Will Initial Records Typically be Loaded\-
6C 61 63 69 70 79 54 20 73 64 72 6F 63 65      01352                  \ in Order\<0><0><0>
69 20 64 65 64 61 6F 4C 20 65 62 20 79 6C      01360
            00 00 00 72 65 64 72 4F 20 6E      0136E
72 6F 63 65 52 20 65 68 74 20 6C 6C 69 57      01378 C.AFP:   .ASCII  \Will the Records be Reloaded Typically i\-
64 65 64 61 6F 6C 65 52 20 65 62 20 73 64      01386                  \n Order\<0>
20 6E 69 20 79 6C 6C 61 63 69 70 79 54 20      01394
                        00 72 65 64 72 4F      013A2
50 20 67 6E 69 64 6E 65 63 73 41 20 79 62      013A8 C.AFQ:   .ASCII  \by Ascending Primary Key\<9>\(Yes/No)[No\-
65 59 28 09 79 65 4B 20 79 72 61 6D 69 72      013B6                  \]\<9>\: \
      20 3A 09 5D 6F 4E 5B 29 6F 4E 2F 73      013C4
69 64 64 41 20 66 6F 20 72 65 62 6D 75 4E      013D0 C.AFR:   .ASCII  \Number of Additional Records to be Added\-
73 64 72 6F 63 65 52 20 6C 61 6E 6F 69 74      013DE                  \ After\<0><0>
41 20 64 65 64 64 41 20 65 62 20 6F 74 20      013EC
                        00 00 72 65 74 66      013FA
69 46 20 6C 61 69 74 69 6E 49 20 65 68 74      01400 C.AFS:   .ASCII  \the Initial File Load\<0><0><0>
            00 00 00 64 61 6F 4C 20 65 6C      0140E
20 67 6E 69 64 61 6F 6C 65 52 20 65 68 74      01418 C.AFT:   .ASCII  \the Reloading the File\<0><0>
            00 00 65 6C 69 46 20 65 68 74      01426
5D 30 5B 29 61 67 69 47 31 2D 30 28 09 09      01430 C.AFU:   .ASCII  <9><9>\(0-1Giga)[0]\<9>\: \<0><0><0>
                        00 00 00 20 3A 09      0143E
72 70 6D 6F 43 20 79 65 4B 20 61 74 61 44      01444 C.AFV:   .ASCII  \Data Key Compression desired\<9>\(Yes/No\-
64 65 72 69 73 65 64 20 6E 6F 69 73 73 65      01452                  \)[Yes]\<9>\: \<0><0><0>
5D 73 65 59 5B 29 6F 4E 2F 73 65 59 28 09      01460
                        00 00 00 20 3A 09      0146E
6F 43 20 64 72 6F 63 65 52 20 61 74 61 44      01474 C.AFW:   .ASCII  \Data Record Compression desired\<9>\(Yes\-
69 73 65 64 20 6E 6F 69 73 73 65 72 70 6D      01482                  \/No)[Yes]\<9>\: \
59 5B 29 6F 4E 2F 73 65 59 28 09 64 65 72      01490
                        20 3A 09 5D 73 65      0149E
73 73 65 72 70 6D 6F 43 20 78 65 64 6E 49      014A4 C.AFX:   .ASCII  \Index Compression desired\<9>\(Yes/No)[Y\-
```

```
59 28 09 64 65 72 69 73 65 64 20 6E 6F 69      014B2                   \es]\<9>\: \<0><0>
20 3A 09 5D 73 65 59 5B 29 6F 4E 2F 73 65      014C0
                        00 00                  014CE
6F 76 20 6B 73 69 64 20 74 65 67 72 61 54      014D0 C.AFY:  .ASCII    \Target disk volume Cluster Size\<9>\(1-1\-
53 20 72 65 74 73 75 6C 43 20 65 6D 75 6C      014DE                   \Giga)[3]\<9>\: \<0>
5B 29 61 67 69 47 31 2D 31 28 09 65 7A 69      014EC
                        00 20 3A 09 5D 33      014FA
70 73 20 6E 61 63 20 73 64 72 6F 63 65 52      01500 C.AFZ:  .ASCII    \Records can span disk blocks\<9>\(Yes/No\-
73 6B 63 6F 6C 62 20 6B 73 69 64 20 6E 61      0150E                   \)[Yes]\<9>\: \<0><0><0>
5D 73 65 59 5B 29 6F 4E 2F 73 65 59 28 09      0151C
                        00 00 00 20 3A 09      0152A
61 6E 6F 69 74 69 64 64 41 20 6C 6C 69 57      01530 C.AGA:  .ASCII    \Will Additional Records Typically be Add\-
69 70 79 54 20 73 64 72 6F 63 65 52 20 6C      0153E                   \ed in\<0><0><0>
64 65 64 64 41 20 65 62 20 79 6C 6C 61 63      0154C
                        00 00 00 6E 69 20      0155A
6E 65 63 73 41 20 79 62 20 72 65 64 72 4F      01560 C.AGB:  .ASCII    \Order by Ascending Primary Key\<9>\(Yes/\-
4B 20 79 72 61 6D 69 72 50 20 67 6E 69 64      0156E                   \No)[No]\<9>\: \<0><0>
6F 4E 5B 29 6F 4E 2F 73 65 59 28 09 79 65      0157C
                        00 00 20 3A 09 5D      0158A
65 56 20 67 6F 6C 6F 72 50 20 65 6C 69 46      01590 C.AGC:  .ASCII    \File Prolog Version\<9><9>\(0-3)[3]\-
33 5B 29 33 2D 30 28 09 09 6E 6F 69 73 72      0159E                   <9>\: \
                        20 3A 09 5D            015AC
                        00 79 65 4B            015B0 C.AGD:  .ASCII    \Key\<0>
      00 00 00 09 09 68 74 67 6E 65 4C 20      015B4 C.AGE:  .ASCII    \ Length\<9><9><0><0><0>
                        00 47 45 53            015C0 C.AGF:  .ASCII    \SEG\<0>
                        00 00 28 09            015C4 C.AGG:  .ASCII    <9>\(\<0><0>
                        00 5D 2D 5B            015C8 C.AGH:  .ASCII    \[-]\<0>
                        00 20 3A 09            015CC C.AGI:  .ASCII    <9>\: \<0>
20 64 65 73 55 20 73 69 73 61 68 70 6D 45      015D0 C.AGJ:  .ASCII    \Emphasis Used In Defining Default:\<9>\(\-
65 44 20 67 6E 69 6E 69 66 65 44 20 6E 49      015DE                   \ \<0><0><0>
      00 00 00 20 28 09 3A 74 6C 75 61 66      015EC
66 75 62 5F 72 65 6C 6C 61 6D 53 20 20 20      015F8 C.AGK:  .ASCII    \   Smaller_buffers )\
                        29 20 73 72 65 66      01606
6C 69 66 5F 72 65 74 74 61 6C 46 20 20 20      0160C C.AGL:  .ASCII    \   Flatter_files   )\
                        29 20 20 20 73 65      0161A
6B 63 75 42 20 64 65 74 73 65 67 67 75 53      01620 C.AGM:  .ASCII    \Suggested Bucket Sizes:\<9><9><9>\( \
20 28 09 09 09 3A 73 65 7A 69 53 20 74 65      0162E
                        00 00 29 20            0163C C.AGN:  .ASCII    \ )\<0><0>
65 76 65 4C 20 66 6F 20 72 65 62 6D 75 4E      01640 C.AGO:  .ASCII    \Number of Levels in Index:\<9><9>\( \
09 09 3A 78 65 64 6E 49 20 6E 69 20 73 6C      0164E
                        20 28                  0165C
                        29 20                  0165E C.AGP:  .ASCII    \ )\
6B 63 75 42 20 66 6F 20 72 65 62 6D 75 4E      01660 C.AGQ:  .ASCII    \Number of Buckets in Index:\<9><9>\( \<0>
09 3A 78 65 64 6E 49 20 6E 69 20 73 74 65      0166E
                        00 20 28 09            0167C
                        00 00 29 20            01680 C.AGR:  .ASCII    \ )\<0><0>
64 65 72 69 75 71 65 52 20 73 65 67 61 50      01684 C.AGS:  .ASCII    \Pages Required to Cache Index:\<9><9>\( \
65 64 6E 49 20 65 68 63 61 43 20 6F 74 20      01692
                        20 28 09 09 3A 78      016A0
                        29 20                  016A6 C.AGT:  .ASCII    \ )\
65 73 55 20 67 6E 69 73 73 65 63 6F 72 50      016A8 C.AGU:  .ASCII    \Processing Used to Search Index:\<9>\( \-
6E 49 20 68 63 72 61 65 53 20 6F 74 20 64      016B6                   <0>
                  00 20 28 09 3A 78 65 64      016C4
                        00 00 29 20            016CC C.AGV:  .ASCII    \ )\<0><0>
                        00 79 65 4B            016D0 C.AGW:  .ASCII    \Key\<0>
09 09 65 7A 69 53 20 74 65 6B 63 75 42 20      016D4 C.AGX:  .ASCII    \ Bucket Size\<9><9>\(\<0>
                        00 28                  016E2
```

```
                                  00 00 00 5B 29 33 36 2D
                                                20 3A 09 5D
65 66 66 75 42 5F 72 65 6C 6C 61 6D 53 28
6C 69 46 5F 72 65 74 74 61 6C 46 20 73 72
                                        00 29 73 65
44 20 72 6F 66 20 73 69 73 61 68 70 6D 45
5F 74 65 6B 63 75 42 20 74 6C 75 61 66 65
5B 29 64 72 6F 77 79 65 4B 28 65 7A 69 53
                                        00 00
                                  20 3A 09 5D 74 61 6C 46
         00 00 00 20 3A 20 5D 6C 6C 61 6D 53
                                        00 79 65 4B
72 65 50 20 6C 6C 69 46 20 64 61 6F 4C 20
5B 29 30 30 31 2D 30 35 28 09 74 6E 65 63
                                  00 20 3A 09 5D 30 30 31
65 20 73 69 68 74 20 65 63 61 6C 70 65 52
64 6E 6F 63 65 73 20 67 6E 69 74 73 69 78
4E 5B 29 6F 4E 2F 73 65 59 28 09 79 72 61
                                  00 20 3A 09 5D 6F
65 6C 69 66 20 65 6C 69 46 20 61 74 61 44
20 36 32 31 2D 31 28 09 09 63 65 70 73 2D
         5D 6C 6C 75 6E 5B 29 73 72 61 68 63
                                        00 00 20 3A
20 65 6C 69 46 20 73 69 73 79 6C 61 6E 41
31 2D 31 28 09 63 65 70 73 2D 65 6C 69 66
6C 6C 75 6E 5B 29 73 72 61 68 63 20 36 32
                                        00 5D
                                     00 00 20 3A
69 66 20 65 6C 69 46 20 74 75 70 74 75 4F
32 31 2D 31 28 09 09 63 65 70 73 2D 65 6C
5D 6C 6C 75 6E 5B 29 73 72 61 68 63 20 36
                                     20 3A
54 20 4C 44 46 20 72 6F 66 20 74 78 65 54
28 09 6E 6F 69 74 63 65 53 20 65 6C 74 69
6E 5B 29 73 72 61 68 63 20 36 32 31 2D 31
                                  5D 6C 6C 75
                                     20 3A
                                  00 79 65 4B
20 32 33 2D 31 28 09 09 09 65 6D 61 4E 20
         5D 6C 6C 75 6E 5B 29 73 72 61 68 63
                                     20 3A
6E 69 42 20 34 6E 69 42 20 32 6E 69 42 28
6E 49 20 34 74 6E 49 20 32 74 6E 49 20 38
72 74 53 20 6C 61 6D 69 63 65 44 20 38 74
                                  00 00 29 67 6E 69
                                  00 79 65 4B
4B 28 09 09 65 70 79 54 20 61 74 61 44 20
3A 09 5D 72 74 53 5B 29 64 72 6F 77 79 65
                                  00 00 00 20
6E 65 72 65 66 65 52 20 66 6F 20 79 65 4B
                                     00 28 09 09 65 63
                                  00 20 3A 09 5D 30 5B 29
73 79 65 4B 20 66 6F 20 72 65 62 6D 75 4E
2D 31 28 09 65 6E 69 66 65 44 20 6F 74 20
                                  00 00 00 5B 29 35 35 32
                                              20 3A 09 5D
75 74 65 52 5F 65 67 61 69 72 72 61 43 28
```

```
016E4 C.AGY:  .ASCII  \-63)[\<0><0><0>
016EC C.AGZ:  .ASCII  \]\<9>\: \
016F0 C.AHA:  .ASCII  \(Smaller_Buffers Flatter_Files)\<0>
016FE
0170C
01710 C.AHB:  .ASCII  \Emphasis for Default Bucket_Size(Keyword\-
0171E                 \)[\<0><0>
0172C
0173A
0173C C.AHC:  .ASCII  \Flat]\<9>\: \
01744 C.AHD:  .ASCII  \Small] : \<0><0><0>
01750 C.AHE:  .ASCII  \Key\<0>
01754 C.AHF:  .ASCII  \ Load Fill Percent\<9>\(50-100)[100]\-
01762                 <9>\: \<0>
01770
01778 C.AHG:  .ASCII  \Replace this existing secondary\<9>\(Yes\-
01786                 \/No)[No]\<9>\: \<0>
01794
017A2
017A8 C.AHH:  .ASCII  \Data File file-spec\<9><9>\(1-126 chars)\-
017B6                 \[null]\
017C4
017D0 C.AHI:  .ASCII  \: \<0><0>
017D4 C.AHJ:  .ASCII  \Analysis File file-spec\<9>\(1-126 chars\-
017E2                 \)[null]\<0>
017F0
017FE
01800 C.AHK:  .ASCII  \: \<0><0>
01804 C.AHL:  .ASCII  \Output File file-spec\<9><9>\(1-126 char\-
01812                 \s)[null]\
01820
0182E C.AHM:  .ASCII  \: \
01830 C.AHN:  .ASCII  \Text for FDL Title Section\<9>\(1-126 ch\-
0183E                 \ars)[null]\
0184C
0185A
0185E C.AHO:  .ASCII  \: \
01860 C.AHP:  .ASCII  \Key\<0>
01864 C.AHQ:  .ASCII  \ Name\<9><9><9>\(1-32 chars)[null]\
01872
0187E C.AHR:  .ASCII  \: \
01880 C.AHS:  .ASCII  \(Bin2 Bin4 Bin8 Int2 Int4 Int8 Decimal S\-
0188E                 \tring)\<0><0>
0189C
018AA
018B0 C.AHT:  .ASCII  \Key\<0>
018B4 C.AHU:  .ASCII  \ Data Type\<9><9>\(Keyword)[Str]\<9>\: \-
018C2                 <0><0><0>
018D0
018D4 C.AHV:  .ASCII  \Key of Reference\<9><9>\(\<0>
018E2
018E8 C.AHW:  .ASCII  \)[0]\<9>\: \<0>
018F0 C.AHX:  .ASCII  \Number of Keys to Define\<9>\(1-255)[\-
018FE                 <0><0><0>
0190C
01914 C.AHY:  .ASCII  \]\<9>\: \
01918 C.AHZ:  .ASCII  \(Carriage_Return FORTRAN None Print)\
```

```
6E 6F 4E 20 4E 41 52 54 52 4F 46 20 6E 72        01926
                  29 74 6E 69 72 50 20 65        01934
72 74 6E 6F 43 20 65 67 61 69 72 72 61 43        0193C C.AIA:  .ASCII  \Carriage Control\<9><9>\(Keyword)[Carr]\-
5B 29 64 72 6F 77 79 65 4B 28 09 09 6C 6F        0194A                 <9>\: \
                  20 3A 09 5D 72 72 61 43        01958
6C 62 61 69 72 61 56 20 64 65 78 69 46 28        01960 C.AIB:  .ASCII  \(Fixed Variable)\
                                  29 65        0196E
6C 62 61 69 72 61 56 20 64 65 78 69 46 28        01970 C.AIC:  .ASCII  \(Fixed Variable VFC)\
                        29 43 46 56 20 65        0197E
20 6D 61 65 72 74 53 20 64 65 78 69 46 28        01984 C.AID:  .ASCII  \(Fixed Stream _CR _LF Undefined Variable\-
69 66 65 64 6E 55 20 46 4C 5F 20 52 43 5F        01992                 \ VFC)\<0><0><05
56 20 65 6C 62 61 69 72 61 56 20 64 65 6E        019A0
                        00 00 00 29 43 46        019AE
09 74 61 6D 72 6F 46 20 64 72 6F 63 65 52        019B4 C.AIE:  .ASCII  \Record Format\<9><9><9>\(Keyword)[Var]\-
61 56 5B 29 64 72 6F 77 79 65 4B 28 09 09        019C2                 <9>\: \<0><0><0>
                  00 00 00 20 3A 09 5D 72        019D0
20 64 6C 65 69 46 20 6C 6F 72 74 6E 6F 43        019D8 C.AIF:  .ASCII  \Control Field Size\<9><9>\(1-\<0>
                  00 2D 31 28 09 09 65 7A 69 53  019E6
                        00 20 3A 09 5D 32 5B 29  019F0 C.AIG:  .ASCII  \)[2]\<9>\: \<0>
                        00 00 00 20 6E 61 65 4D  019F8 C.AIH:  .ASCII  \Mean \<0><0><0>
         00 65 7A 69 53 20 64 72 6F 63 65 52     01A00 C.AII:  .ASCII  \Record Size\<0>
                        00 00 78 69 66 2F 77 20  01A0C C.AIJ:  .ASCII  \ w/fix\<0><0>
                        00 00 00 2D 31 28 09 09  01A14 C.AIK:  .ASCII  <9><9>\(1-\<0><0><0>
                                    00 5D 2D 5B  01A1C C.AIL:  .ASCII  \[-]\<0>
                                    00 20 3A 09  01A20 C.AIM:  .ASCII  <9>\: \<0>
                                    00 00 09 09  01A24 C.AIN:  .ASCII  <9><9><0><0>
                                    79 65 4B 20  01A28 C.AIO:  .ASCII  \ Key\
65 53 20 65 70 79 54 20 68 70 61 72 47 20        01A2C C.AIP:  .ASCII  \ Graph Type Selection \<0><0>
                  00 00 20 6E 6F 69 74 63 65 6C  01A3A
69 53 20 74 65 6B 63 75 42 09 65 6E 69 4C        01A44 C.AIQ:  .ASCII  \Line\<9>\Bucket Size vs Index Depth\-
65 44 20 78 65 64 6E 49 20 73 76 20 65 7A        01A52                 <9>\as a 2 dimensional plot\<0>
6D 69 64 20 32 20 61 20 73 61 09 68 74 70        01A60
00 74 6F 6C 70 20 6C 61 6E 6F 69 73 6E 65        01A6E
69 53 20 74 65 6B 63 75 42 09 6C 6C 69 46        01A7C C.AIR:  .ASCII  \Fill\<9>\Bucket Size vs     Load Fill Pe\-
64 61 6F 4C 20 20 20 20 20 73 76 20 65 7A        01A8A                 \rcent     vs Index Depth\
20 74 6E 65 63 72 65 50 20 6C 6C 69 46 20        01A98
44 20 78 65 64 6E 49 20 73 76 20 20 20 20        01AA6
                              68 74 70 65        01AB4
7A 69 53 20 74 65 6B 63 75 42 09 79 65 4B        01AB8 C.AIS:  .ASCII  \Key\<9>\Bucket Size vs          Key Lengt\-
4B 20 20 20 20 20 20 20 20 20 73 76 20 65        01AC6                 \h         vs Index Depth\<0>
20 20 20 20 20 68 74 67 6E 65 4C 20 79 65        01AD4
65 44 20 78 65 64 6E 49 20 73 76 20 20 20        01AE2
                              00 68 74 70        01AF0
20 74 65 6B 63 75 42 09 64 72 6F 63 65 52        01AF4 C.AIT:  .ASCII  \Record\<9>\Bucket Size vs        Record \-
20 20 20 20 20 20 20 73 76 20 65 7A 69 53        01B02                 \Size        vs Index Depth\<0><0>
20 20 65 7A 69 53 20 64 72 6F 63 65 52 20        01B10
78 65 64 6E 49 20 73 76 20 20 20 20 20 20        01B1E
                  00 00 68 74 70 65 44 20        01B2C
69 53 20 74 65 6B 63 75 42 09 74 69 6E 49        01B34 C.AIU:  .ASCII  \Init\<9>\Bucket Size vs Initial Load Rec\-
20 6C 61 69 74 69 6E 49 20 73 76 20 65 7A        01B42                 \ord Count vs Index Depth\
6F 43 20 64 72 6F 63 65 52 20 64 61 6F 4C        01B50
44 20 78 65 64 6E 49 20 73 76 20 74 6E 75        01B5E
                              68 74 70 65        01B6C
7A 69 53 20 74 65 6B 63 75 42 09 64 64 41        01B70 C.AIV:  .ASCII  \Add\<9>\Bucket Size vs  Additional Recor\-
6E 6F 69 74 69 64 64 41 20 20 73 76 20 65        01B7E                 \d Count  vs Index Depth\<0>
6E 75 6F 43 20 64 72 6F 63 65 52 20 6C 61        01B8C
65 44 20 78 65 64 6E 49 20 73 76 20 20 74        01B9A
```

```
                                                  00 68 74 70     01BA8
79 65 4B 20 6C 6C 69 46 20 65 6E 69 4C 28         01BAC C.AIW:  .ASCII  \(Line Fill Key\<0><0>
                                  00 00           01BBA
41 20 74 69 6E 49 20 64 72 6F 63 65 52 20         01BBC C.AIX:  .ASCII  \ Record Init Add)\<0><0><0>
                     00 00 00 29 64 64            01BCA
20 6F 74 20 65 70 79 74 20 68 70 61 72 47         01BD0 C.AIY:  .ASCII  \Graph type to display\<9><9>\(Keyword)[\-
77 79 65 4B 28 09 09 79 61 6C 70 73 69 64         01BDE                 <0><0><0>
                  00 00 00 5B 29 64 72 6F         01BEC
                  20 3A 09 5D 65 6E 69 4C         01BF4 C.AIZ:  .ASCII  \Line]\<9>\: \
                  20 3A 09 5D 6C 6C 69 46         01BFC C.AJA:  .ASCII  \Fill]\<9>\: \
                  00 20 3A 09 5D 79 65 4B         01C04 C.AJB:  .ASCII  \Key]\<9>\: \<0>
                  00 20 3A 09 5D 63 65 52         01C0C C.AJC:  .ASCII  \Rec]\<9>\: \<0>
                  20 3A 09 5D 74 69 6E 49         01C14 C.AJD:  .ASCII  \Init]\<9>\: \
                  00 20 3A 09 5D 64 64 41         01C1C C.AJE:  .ASCII  \Add]\<9>\: \<0>
                                  00 00 09 09     01C24 C.AJF:  .ASCII  <9><9><0><0>
72 61 6C 75 6E 61 72 47 20 61 65 72 41 20         01C28 C.AJG:  .ASCII  \ Area Granularity Selection \
20 6E 6F 69 74 63 65 6C 65 53 20 79 74 69         01C36
71 71 71 71 71 71 6C 0E 30 29 1B 20 20 20         01C44 C.AJH:  .ASCII  \   \<27>\)0\<14>\lqqqqqqqqqqqqqqk\<15>\  \-
20 20 20 20 20 0F 6B 71 71 71 71 71 71 71         01C52                 \   \<14>\lqqqqqqqqqqqqqqk\<15>\     \-
71 71 71 71 71 71 71 71 71 71 71 71 6C 0E         01C60                 <14>\lqqqqqqqqqqqqqqk\<15>\    \<14>\lqqq\
71 71 71 71 6C 0E 20 20 20 20 20 0F 6B 71         01C6E
20 20 20 0F 6B 71 71 71 71 71 71 71 71 71         01C7C
      71 71 71 71 71 71 71 71 6C 0E 20 20         01C8A
                              00 00 20 0F         01C96         .ASCII  <15>\ \<0><0>
44 20 30 20 79 65 4B 20 0F 78 0E 20 30 20         01CA0 C.AJI:  .ASCII  \ 0 \<14>\x\<15>\ Key 0 Data  \<14>\x\-
0E 20 30 20 20 20 0F 78 0E 20 20 61 74 61         01CAE                 <15>\   0 \<14>\x\<15>\ Key 0 Data  \-
20 61 74 61 44 20 30 20 79 65 4B 20 0F 78         01CBC                 <14>\x\<15>\   0 \<14>\x\<15>\ Key 0 Data\-
4B 20 0F 78 0E 20 30 20 20 20 0F 78 0E 20         01CCA                 \  \<14>\x\<15>\   0 \<14>\x\<15>
0F 78 0E 20 20 61 74 61 44 20 30 20 79 65         01CD8
      79 65 4B 20 0F 78 0E 20 30 20 20 20         01CE6
                        00 20 0F 78 0E            01CF2         .ASCII  <14>\x\<15>\ \<0>
20 20 20 20 20 20 20 20 0F 78 0E 20 20 20         01D00 C.AJJ:  .ASCII  \   \<14>\x\<15>\              \<14>\x\-
0E 20 20 20 20 20 0F 78 0E 20 20 20 20 20         01D0E                 <15>\     \<14>\tqqqqqqqqqqqqqqu\<15>\    \-
71 71 71 71 71 71 71 71 71 71 71 71 74            01D1C                 \ \<14>\tqqqqqqqqqqqqqqu\<15>\     \<14>\t\
71 71 71 71 71 74 0E 20 20 20 20 20 0F 75         01D2A
20 20 20 20 0F 75 71 71 71 71 71 71 71 71         01D38
      71 71 71 71 71 71 71 71 71 74 0E 20         01D46
                              00 00 00 20 0F      01D52         .ASCII  <15>\ \<0><0><0>
49 20 30 20 79 65 4B 20 0F 78 0E 20 20 20         01D5C C.AJK:  .ASCII  \   \<14>\x\<15>\ Key 0 Index \<14>\x\-
0E 20 31 20 20 20 0F 78 0E 20 78 65 64 6E         01D6A                 <15>\   1 \<14>\x\<15>\ Key 0 Index \-
78 65 64 6E 49 20 30 20 79 65 4B 20 0F 78         01D78                 <14>\x\<15>\   1 \<14>\x\<15>\ Key 0 Inde\-
4B 20 0F 78 0E 20 31 20 20 20 0F 78 0E 20         01D86                 \x \<14>\x\<15>\   1 \<14>\x\<15>
0F 78 0E 20 78 65 64 6E 49 20 30 20 79 65         01D94
      79 65 4B 20 0F 78 0E 20 31 20 20 20         01DA2
                        00 20 0F 78 0E            01DAE         .ASCII  <14>\x\<15>\ \<0>
20 20 20 20 20 20 20 20 0F 78 0E 20 20 20         01DBC C.AJL:  .ASCII  \   \<14>\x\<15>\              \<14>\x\-
0E 20 20 20 20 20 0F 78 0E 20 20 20 20 20         01DCA                 <15>\     \<14>\x\<15>\              \-
20 20 20 20 20 20 20 20 20 20 20 20 0F 78         01DD8                 <14>\x\<15>\     \<14>\tqqqqqqqqqqqqqqu\-
71 71 71 74 0E 20 20 20 20 20 0F 78 0E 20         01DE6                 <15>\     \<14>
20 20 0F 75 71 71 71 71 71 71 71 71 71 71         01DF4
      71 71 71 71 71 71 71 74 0E 20 20 20         01E02
                              00 20 0F            01E0E         .ASCII  <15>\ \<0>
44 20 6E 20 79 65 4B 20 0F 78 0E 20 20 20         01E18 C.AJM:  .ASCII  \   \<14>\x\<15>\ Key n Data  \<14>\x\-
0E 20 20 20 20 20 0F 78 0E 20 20 61 74 61         01E26                 <15>\     \<14>\x\<15>\ Key n Data  \-
20 61 74 61 44 20 6E 20 79 65 4B 20 0F 78         01E34                 <14>\x\<15>\   2 \<14>\x\<15>\ Key n Data\-
4B 20 0F 78 0E 20 32 20 20 20 0F 78 0E 20         01E42                 \  \<14>\x\<15>\   2 \<14>\x\<15>
0F 78 0E 20 20 61 74 61 44 20 6E 20 79 65         01E50
```

```
                79 65 4B 20 0F 78 0E 20 32 20 20 20      01E5E
                                     00 20 0F 78 0E      01E6A
20 20 20 20 20 20 20 20 0F 78 0E 20 20 20      .ASCII  <14>\x\<15>\ \<0>
0E 20 20 20 20 20 0F 78 0E 20 20 20 20 20      01E78 C.AJN:  .ASCII  \   \<14>\x\<15>\              \<14>\x\-
20 20 20 20 20 20 20 20 20 20 20 20 0F 78      01E86                 <15>\     \<14>\x\<15>\              \-
20 20 0F 78 0E 20 20 20 20 20 0F 78 0E 20      01E94                 <14>\x\<15>\     \<14>\x\<15>\          \-
0F 78 0E 20 20 20 20 20 20 20 20 20 20 20      01EA2                 \  \<14>\x\<15>\     \<14>
                71 71 71 71 71 74 0E 20 20 20 20 20      01EB0
                                     00 00 00 20 0F      01EBE
49 20 6E 20 79 65 4B 20 0F 78 0E 20 20 20      01ECA         .ASCII  <15>\ \<0><0><0>
0E 20 20 20 20 20 0F 78 0E 20 78 65 64 6E      01ED8 C.AJO:  .ASCII  \   \<14>\x\<15>\ Key n Index \<14>\x\-
78 65 64 6E 49 20 6E 20 79 65 4B 20 0F 78      01EE6                 <15>\     \<14>\x\<15>\ Key n Index \-
4B 20 0F 78 0E 20 20 20 20 20 0F 78 0E 20      01EF4                 <14>\x\<15>\     \<14>\x\<15>\ Key n Inde\-
0F 78 0E 20 78 65 64 6E 49 20 6E 20 79 65      01F02                 \x \<14>\x\<15>\   3 \<14>\x\<15>
                79 65 4B 20 0F 78 0E 20 33 20 20 20      01F10
                                     00 20 0F 78 0E      01F1E
71 71 71 71 71 71 71 71 71 6D 0E 20 20 20      01F2A         .ASCII  <14>\x\<15>\ \<0>
71 6D 0E 20 20 20 20 20 0F 6A 71 71 71 71      01F38 C.AJP:  .ASCII  \   \<14>\mqqqqqqqqqqqqqj\<15>\     \-
0F 6A 71 71 71 71 71 71 71 71 71 71 71 71      01F46                 <14>\mqqqqqqqqqqqqqj\<15>\     \<14>\mqqq\-
71 71 71 71 71 71 71 6D 0E 20 20 20 20 20      01F54                 \qqqqqqqqqqj\<15>\     \<14>\mqqqqqqqqqqq\
0E 20 20 20 20 20 0F 6A 71 71 71 71 71 71      01F62
                71 71 71 71 71 71 71 71 71 71 71 6D      01F70
                                     00 20 0F      01F7E
29 31 28 20 65 6E 4F 20 20 20 20 20 20 20      01F8A         .ASCII  <15>\ \<0>
54 20 20 20 20 20 20 20 20 20 20 20 20 20      01F90 C.AJQ:  .ASCII  \       One (1)            Two (2)      \-
20 20 20 20 20 20 20 20 20 29 32 28 20 6F      01F9E                 \     Three (3)            Four (4)\<0>
20 29 33 28 20 65 65 72 68 54 20 20 20 20      01FAC
75 6F 46 20 20 20 20 20 20 20 20 20 20 20      01FBA
                                     00 29 34 28 20 72      01FC8
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20 20 20      01FD6
2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D 2D 2D      01FDC C.AJR:  .ASCII  \   +-------------+      +-------------+  \-
20 20 20 20 2B 2D 2D 2D 2D 2D 2D 2D 2D 2D      01FEA                 \   +-------------+      +-------------+\-
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20      01FF8                 <0><0>
2D 2D 2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D      02006
                00 00 2B 2D 2D 2D 2D 2D 2D 2D 2D      02014
74 61 44 20 30 20 79 65 4B 20 7C 20 30 20      02022
79 65 4B 20 7C 20 30 20 20 20 7C 20 20 61      0202C C.AJS:  .ASCII  \ 0 ¦ Key 0 Data  ¦   0 ¦ Key 0 Data  ¦  \-
30 20 20 20 7C 20 20 61 74 61 44 20 30 20      0203A                 \ 0 ¦ Key 0 Data  ¦   0 ¦ Key 0 Data  ¦\-
20 61 74 61 44 20 30 20 79 65 4B 20 7C 20      02048                 <0><0>
30 20 79 65 4B 20 7C 20 30 20 20 20 7C 20      02056
                00 00 7C 20 20 61 74 61 44 20      02064
20 20 20 20 20 20 20 20 20 20 7C 20 20 20      02072
2D 2D 2D 2D 2B 20 20 20 20 20 7C 20 20 20      0207C C.AJT:  .ASCII  \   ¦             ¦      +-------------+  \-
20 20 20 20 2B 2D 2D 2D 2D 2D 2D 2D 2D 2D      0208A                 \   +-------------+      +-------------+\-
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20      02098                 <0><0>
2D 2D 2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D      020A6
                00 00 2B 2D 2D 2D 2D 2D 2D 2D 2D      020B4
64 6E 49 20 30 20 79 65 4B 20 7C 20 20 20      020C2
79 65 4B 20 7C 20 31 20 20 20 7C 20 78 65      020CC C.AJU:  .ASCII  \   ¦ Key 0 Index ¦   1 ¦ Key 0 Index ¦  \-
31 20 20 20 7C 20 78 65 64 6E 49 20 30 20      020DA                 \ 1 ¦ Key 0 Index ¦   1 ¦ Key 0 Index ¦\-
78 65 64 6E 49 20 30 20 79 65 4B 20 7C 20      020E8                 <0><0>
30 20 79 65 4B 20 7C 20 31 20 20 20 7C 20      020F6
                00 00 7C 20 78 65 64 6E 49 20      02104
20 20 20 20 20 20 20 20 20 20 7C 20 20 20      02112
20 20 20 20 7C 20 20 20 20 20 7C 20 20 20      0211C C.AJV:  .ASCII  \   ¦             ¦      ¦             ¦  \-
20 20 20 20 7C 20 20 20 20 20 20 20 20 20      0212A                 \   +-------------+      +-------------+\-
20 20 20 20 7C 20 20 20 20 20 20 20 20 20      02138                 <0><0>
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20      02146
```

```
2D 2D 2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D        02154
                00 00 2B 2D 2D 2D 2D 2D 2D 2D    02162
74 61 44 20 6E 20 79 65 4B 20 7C 20 20 20        0216C C.AJW:  .ASCII  \   ¦ Key n Data ¦     ¦ Key n Data ¦  \-
79 65 4B 20 7C 20 20 20 20 20 7C 20 20 61        0217A                 \ 2 ¦ Key n Data ¦   2 ¦ Key n Data ¦\-
32 20 20 20 7C 20 20 61 74 61 44 20 6E 20        02188                 <0><0>
20 61 74 61 44 20 6E 20 79 65 4B 20 7C 20        02196
6E 20 79 65 4B 20 7C 20 32 20 20 20 7C 20        021A4
                00 00 7C 20 20 61 74 61 44 20    021B2
20 20 20 20 20 20 20 20 20 20 7C 20 20 20        021BC C.AJX:  .ASCII  \   ¦                 ¦     ¦             ¦  \-
20 20 20 20 7C 20 20 20 20 20 7C 20 20 20        021CA                 \   ¦                 ¦     +-------------+\-
20 20 20 20 7C 20 20 20 20 20 20 20 20 20        021D8                 <0><0>
20 20 20 20 20 20 20 20 20 20 20 20 7C 20        021E6
2D 2D 2D 2D 2D 2D 2B 20 20 20 20 20 7C 20        021F4
                00 00 2B 2D 2D 2D 2D 2D 2D 2D    02202
64 6E 49 20 6E 20 79 65 4B 20 7C 20 20 20        0220C C.AJY:  .ASCII  \   ¦ Key n Index ¦     ¦ Key n Index ¦  \-
79 65 4B 20 7C 20 20 20 20 20 7C 20 78 65        0221A                 \   ¦ Key n Index ¦   3 ¦ Key n Index ¦\-
20 20 20 20 7C 20 78 65 64 6E 49 20 6E 20        02228                 <0><0>
78 65 64 6E 49 20 6E 20 79 65 4B 20 7C 20        02236
6E 20 79 65 4B 20 7C 20 33 20 20 20 7C 20        02244
                00 00 7C 20 78 65 64 6E 49 20    02252
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20 20 20        0225C C.AJZ:  .ASCII  \   +-------------+     +-------------+  \-
2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D 2D 2D        0226A                 \   +-------------+     +-------------+\-
20 20 20 20 2B 2D 2D 2D 2D 2D 2D 2D 2D 2D        02278                 <0><0>
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B 20 20        02286
2D 2D 2D 2D 2D 2D 2B 20 20 20 20 20 2B 2D        02294
                00 00 2B 2D 2D 2D 2D 2D 2D 2D    022A2
29 31 28 20 65 6E 4F 20 20 20 20 20 20 20        022AC C.AKA:  .ASCII  \       One (1)           Two (2)       \-
54 20 20 20 20 20 20 20 20 20 20 20 20 20        022BA                 \     Three (3)           Four (4)\<0>
20 20 20 20 20 20 20 20 29 32 28 20 6F 77        022C8
20 29 33 28 20 65 65 72 68 54 20 20 20 20        022D6
75 6F 46 20 20 20 20 20 20 20 20 20 20 20        022E4
                00 29 34 28 20 72                022F2
65 65 72 68 54 20 6F 77 54 20 65 6E 4F 28        022F8 C.AKB:  .ASCII  \(One Two Three Four Double)\<0>
00 29 65 6C 62 75 6F 44 20 72 75 6F 46 20        02306
22 65 6C 62 75 6F 44 22 20 65 70 79 54 28        02314 C.AKC:  .ASCII  \(Type "Double" to allocate 2 areas per k\-
32 20 65 74 61 63 6F 6C 6C 61 20 6F 74 20        02322                 \ey)\<0>
79 65 6B 20 72 65 70 20 73 61 65 72 61 20        02330
                                00 29            0233E
61 65 72 61 20 66 6F 20 72 65 62 6D 75 4E        02340 C.AKD:  .ASCII  \Number of areas to allocate\<9>\(keyword\-
09 65 74 61 63 6F 6C 6C 61 20 6F 74 20 73        0234E                 \)[Three] : \<0>
65 72 68 54 5B 29 64 72 6F 77 79 65 6B 28        0235C
                        00 20 3A 20 5D 65        0236A
                              00 00 09 09        02370 C.AKE:  .ASCII  <9><9><0><0>
45 53 20 72 6F 74 69 64 45 20 4C 44 46 20        02374 C.AKF:  .ASCII  \ FDL Editor SET Function \<0><0><0>
00 00 00 20 6E 6F 69 74 63 6E 75 46 20 54        02382
73 65 6C 69 66 09 73 69 73 79 6C 61 6E 41        02390 C.AKG:  .ASCII  \Analysis\<9>\filespec of FDL Analysis fi\-
61 6E 41 20 4C 44 46 20 66 6F 20 63 65 70        0239E                 \le\<0><0>
      00 00 65 6C 69 66 20 73 69 73 79 6C        023AC
20 65 70 79 74 09 09 79 61 6C 70 73 69 44        023B8 C.AKH:  .ASCII  \Display\<9><9>\type of graph to display\-
69 64 20 6F 74 20 68 70 61 72 67 20 66 6F        023C6                 <0><0><0>
                  00 00 00 79 61 6C 70 73        023D4
65 64 20 66 6F 09 73 69 73 61 68 70 6D 45        023DC C.AKI:  .ASCII  \Emphasis\<9>\of default bucketsize calcu\-
69 73 74 65 6B 63 75 62 20 74 6C 75 61 66        023EA                 \lations\<0>
6E 6F 69 74 61 6C 75 63 6C 61 63 20 65 7A        023F8
                                00 73            02406
75 6E 09 79 74 69 72 61 6C 75 6E 61 72 47        02408 C.AKJ:  .ASCII  \Granularity\<9>\number of areas in Index\-
20 73 61 65 72 61 20 66 6F 20 72 65 62 6D        02416                 \ed files\
```

```
                                                
```

```
65 20 65 68 74 20 74 63 65 66 66 61 20 73      0269A                   \S data file\<0>
61 74 61 64 20 53 4D 52 20 65 72 69 74 6E      026A8
                        00 65 6C 69 66 20      026B6
62 69 72 74 74 61 09 4C 41 4E 52 55 4F 4A      026BC C.ALG:  .ASCII    \JOURNAL\<9>\attributes set the journalin\-
6A 20 65 68 74 20 74 65 73 20 73 65 74 75      026CA                   \g parameters of the file\
61 72 61 70 20 67 6E 69 6C 61 6E 72 75 6F      026D8
20 65 68 74 20 66 6F 20 73 72 65 74 65 6D      026E6
                        74 75 62 69 72 74      026F4
74 75 62 69 72 74 74 61 09 79 20 59 45 4B      026F8 C.ALH:  .ASCII    \KEY y\<9>\attributes define the characte\-
20 65 68 74 20 65 6E 69 66 65 64 20 73 65      02706                   \ristics of key y\
63 69 74 73 69 72 65 74 63 61 72 61 68 63      02714
                        79 20 79 65 6B 20      02722
75 62 69 72 74 74 61 09 44 52 4F 43 45 52      0272C C.ALI:  .ASCII    \RECORD\<9>\attributes set the non-key as\-
6F 6E 20 65 68 74 20 74 65 73 20 73 65 74      0273A                   \pects of each record\
20 73 74 63 65 70 73 61 20 79 65 6B 2D 6E      02748
64 72 6F 63 65 72 20 68 63 61 65 20 66 6F      02756
62 69 72 74 74 61 09 47 4E 49 52 41 48 53      02764 C.ALJ:  .ASCII    \SHARING\<9>\attributes set the run-time \-
72 20 65 68 74 20 74 65 73 20 73 65 74 75      02772                   \sharing mode of the file\
6E 69 72 61 68 73 20 65 6D 69 74 2D 6E 75      02780
20 65 68 74 20 66 6F 20 65 64 6F 6D 20 67      0278E
                        65 6C 69 66 20 65      0279C
75 62 69 72 74 74 61 09 4D 45 54 53 59 53      027A0 C.ALK:  .ASCII    \SYSTEM\<9>\attributes document operating\-
6F 20 74 6E 65 6D 75 63 6F 64 20 73 65 74      027AE                   \ system-specific items\<0><0>
65 74 73 79 73 20 67 6E 69 74 61 72 65 70      027BC
65 74 69 20 63 69 66 69 63 65 70 73 2D 6D      027CA
                        00 00 73 6D 65 20      027D8
68 20 65 68 74 20 73 69 09 45 4C 54 49 54      027DC C.ALL:  .ASCII    \TITLE\<9>\is the header line for the FDL\-
72 6F 66 20 65 6E 69 6C 20 72 65 64 61 65      027EA                   \ file\<0><0><0>
00 65 6C 69 66 20 4C 44 46 20 65 68 74 20      027F8
                        00 00 73 6D 65 20      02806
43 20 41 45 52 41 20 53 53 45 43 43 41 28      02808 C.ALM:  .ASCII    \(ACCESS AREA CONNECT DATE FILE JOURNAL\-
49 46 20 45 54 41 44 20 54 43 45 4E 4E 4F      02816                   <0><0>
         00 00 4C 41 4E 52 55 4F 4A 20 45 4C   02824
48 53 20 44 52 4F 43 45 52 20 59 45 4B 20      02830 C.ALN:  .ASCII    \ KEY RECORD SHARING SYSTEM TITLE)\<0>
54 20 4D 45 54 53 59 53 20 47 4E 49 52 41      0283E
                        00 29 45 4C 54 49      0284C
                        09 09                  02852 C.ALO:  .ASCII    <9><9>
61 6D 69 72 50 20 74 6E 65 72 72 75 43 20      02854 C.ALP:  .ASCII    \ Current Primary Attributes \
20 73 65 74 75 62 69 72 74 74 41 20 79 72      02862
20 72 6F 66 2C 22 3F 22 20 65 70 79 54 28      02870 C.ALQ:  .ASCII    \(Type "?" for a list of existing Primary\-
73 69 78 65 20 66 6F 20 74 73 69 6C 20 61      0287E                   \ Attributes)\
41 20 79 72 61 6D 69 72 50 20 67 6E 69 74      0288C
                        29 73 65 74 75 62      0289A
20 64 65 72 69 73 65 44 20 72 65 74 6E 45      028A4 C.ALR:  .ASCII    \Enter Desired Primary\<9><9>\(Keyword)[\-
77 79 65 4B 22 09 09 79 72 61 6D 69 72 50      028B2                   <0><0><0>
                        00 00 00 5B 29 64 72 6F 028C0
        0000006B  00000064  00000055  00000010  00000048  028C8 C.ALS:  .LONG    72,16,85,100,107,111,128,144,149,157,162,-
        0000009D  00000095  00000090  00000080  0000006F  028DC                 168,174,182,186,194,202,209
        000000BA  000000B6  000000AE  000000A8  000000A2  028F0
                            000000D1  000000CA  000000C2  02904
    45 50 59 54 5F 59 52 41 4D 49 52 50 0C     02910         .ASCII    <12>\PRIMARY_TYPE\
59 52 41 4D 49 52 50 5F 59 4D 4D 55 44 0E      0291D         .ASCII    <14>\DUMMY_PRIMARY$\
                                        24     0292B
                        53 53 45 43 43 41 06   0292C         .ASCII    <6>\ACCESS\
                              4C 43 41 03      02933         .ASCII    <3>\ACL\
41 5F 46 4F 5F 53 49 53 59 4C 41 4E 41 10      02937         .ASCII    <16>\ANALYSIS_OF_AREA\
                              41 45 52         02945
```

```
 4B 5F 46 4F 5F 53 49 53 59 4C 41 4E 41 0F      02948           .ASCII  <15>\ANALYSIS_OF_KEY\
                                    59 45      02956
                                 41 45 52 41 04 02958           .ASCII  <4>\AREA\
                              54 43 45 4E 4E 4F 43 07 0295D     .ASCII  <7>\CONNECT\
                                    45 54 41 44 04 02965     .ASCII  <4>\DATE\
                                 24 45 4C 49 46 05 0296A     .ASCII  <5>\FILE$\
                                 54 4E 45 44 49 05 02970     .ASCII  <5>\IDENT\
                              4C 41 4E 52 55 4F 4A 07 02976     .ASCII  <7>\JOURNAL\
                                    59 45 4B 03 0297E     .ASCII  <3>\KEY\
                              24 44 52 4F 43 45 52 07 02982     .ASCII  <7>\RECORD$\
                              47 4E 49 52 41 48 53 07 0298A     .ASCII  <7>\SHARING\
                                 4D 45 54 53 59 53 06 02992     .ASCII  <6>\SYSTEM\
                                 00 45 4C 54 49 54 05 02999     .ASCII  <5>\TITLE\<0>
                                          08 20 029A0 C.ALT:    .BYTE   ^X20,8
                                                029A2           .BLKB   2
                                    20 3A 20 5D 029A4 C.ALU:    .ASCII  \] : \
    00000000 00000000 00000000 00000000 00070000 029A8 C.ALV:   .LONG   ^X70000,0,0,0,0,0,0,0
             00000000 00000000 00000000 00000000 029BC
    00000000 00000000 00000000 04410000 00000000 029C8 C.ALW:   .LONG   0,^X4410000,0,0,0,0,0,0
             00000000 00000000 00000000 00000000 029DC
    00000000 00000000 00000000 00000000 00070000 029E8 C.ALX:   .LONG   ^X70000,0,0,0,0,0,0,0
             00000000 00000000 00000000 00000000 029FC
    00000000 00000000 00000000 00000000 00000031 02A08 C.ALY:   .LONG   ^X31,0,0,0,0,0,0,0
             00000000 00000000 00000000 00000000 02A1C
                                    3B 70 50 1B 02A28 C.ALZ:    .ASCII  <27>\Pp;\
28 57 28 54 3B 5D 35 38 32 2C 37 32 5B 50 02A2C C.AMA:   .ASCII  \P[27,285];T(W(I3))' Key:    Good   ';T(W\-
20 20 20 20 3A 79 65 4B 20 27 29 29 33 49 02A3A                   \(I2))'Fair   ';T(W(I1))'Poor';\<0><0>
49 28 57 28 54 3B 27 20 20 20 64 6F 6F 47 02A48
54 3B 27 20 20 20 72 69 61 46 27 29 29 32 02A56
3B 27 72 6F 6F 50 27 29 29 31 49 28 57 28 02A64
                                    00 00 02A72
28 57 28 54 3B 5D 30 32 33 2C 37 32 5B 50 02A74 C.AMB:   .ASCII  \P[27,320];T(W(I3))'\<0>
                                 00 27 29 29 33 49 02A82
73 72 65 56 20 67 6F 6C 6F 72 50 2D 56 50 02A88 C.AMC:   .ASCII  \PV-Prolog Version       \
             20 20 20 20 20 20 20 20 6E 6F 69 02A96
             00 00 79 65 4B 2D 54 4B 02AA0 C.AMD:   .ASCII  \KT-Key\<0><0>
       00 00 00 20 20 20 20 65 70 79 54 20 02AA8 C.AME:   .ASCII  \ Type    \<0><0><0>
                   20 32 6E 69 42 20 20 20 02AB4 C.AMF:   .ASCII  \   Bin2 \
                   20 34 6E 69 42 20 20 20 02ABC C.AMG:   .ASCII  \   Bin4 \
                   20 38 6E 69 42 20 20 20 02AC4 C.AMH:   .ASCII  \   Bin8 \
                   20 6C 61 6D 69 63 65 44 02ACC C.AMI:   .ASCII  \Decimal \
                   20 32 74 6E 49 20 20 20 02AD4 C.AMJ:   .ASCII  \   Int2 \
                   20 34 74 6E 49 20 20 20 02ADC C.AMK:   .ASCII  \   Int4 \
                   20 38 74 6E 49 20 20 20 02AE4 C.AML:   .ASCII  \   Int8 \
                   20 67 6E 69 72 74 53 20 02AEC C.AMM:   .ASCII  \ String \
00 20 20 73 69 73 61 68 70 6D 45 2D 4D 45 02AF4 C.AMN:   .ASCII  \EM-Emphasis  \<0><0><0>
                                    00 00 02B02
                   00 00 00 20 20 20 20 20 02B04 C.AMO:   .ASCII  \     \<0><0><0>
                   00 72 65 6C 6C 61 6D 53 02B0C C.AMP:   .ASCII  \Smaller\<0>
                   00 72 65 74 74 61 6C 46 02B14 C.AMQ:   .ASCII  \Flatter\<0>
                                 00 00 28 20 02B1C C.AMR:   .ASCII  \ (\<0><0>
       00 00 79 65 4B 20 70 75 44 2D 4B 44 02B20 C.AMS:   .ASCII  \DK-Dup Key\<0><0>
       00 00 00 20 20 73 65 75 6C 61 56 20 02B2C C.AMT:   .ASCII  \ Values  \<0><0><0>
                                 20 73 65 59 02B38 C.AMU:   .ASCII  \Yes \
                                 20 6F 4E 20 02B3C C.AMV:   .ASCII  \ No \
                   00 00 79 65 4B 2D 4C 4B 02B40 C.AMW:   .ASCII  \KL-Key\<0><0>
00 20 20 20 20 20 20 68 74 67 6E 65 4C 20 02B48 C.AMX:   .ASCII  \ Length      \<0><0><0>
```

```
                                                                  00 00     02B56
                                                  00 00 79 65 4B 2D 50 4B   02B58 C.AMY:   .ASCII  \KP-Key\<0><0>
                                  00 20 20 6E 6F 69 74 69 73 6F 50 20       02B60 C.AMZ:   .ASCII  \ Position  \<0>
                              64 72 6F 63 65 52 20 61 74 61 44 2D 43 52     02B6C C.ANA:   .ASCII  \RC-Data Record Comp  \<0>
                                                  00 20 20 70 6D 6F 43 20   02B7A
                                                                  20 25     02B82 C.ANB:   .ASCII  \% \
                              6F 43 20 79 65 4B 20 61 74 61 44 2D 43 4B     02B84 C.ANC:   .ASCII  \KC-Data Key Comp     \<0>
                                                  00 20 20 20 20 20 70 6D   02B92
                                                                  20 25     02B9A C.AND:   .ASCII  \% \
                              72 6F 63 65 52 20 78 65 64 6E 49 2D 43 49     02B9C C.ANE:   .ASCII  \IC-Index Record Comp \<0>
                                                  00 20 70 6D 6F 43 20 64   02BAA
                                                                  20 25     02BB2 C.ANF:   .ASCII  \% \
                              6C 6C 69 46 20 74 65 6B 63 75 42 2D 46 42     02BB4 C.ANG:   .ASCII  \BF-Bucket Fill       \<0>
                                                  00 20 20 20 20 20 20 20   02BC2
                                                                  20 25     02BCA C.ANH:   .ASCII  \% \
                              6D 72 6F 46 20 64 72 6F 63 65 52 2D 46 52     02BCC C.ANI:   .ASCII  \RF-Record Format \<0><0><0>
                                                  00 00 00 20 74 61 64 6F   02BDA
                                  00 00 00 20 65 6C 62 61 69 72 61 56       02BE0 C.ANJ:   .ASCII  \Variable \<0><0><0>
                                  00 00 00 20 64 65 78 69 46 20 20 20       02BEC C.ANK:   .ASCII  \   Fixed \<0><0><0>
                                                  00 2D 53 52 65 7A 69 7A   02BF8 C.ANL:   .ASCII  \RS-\<0>
                              69 53 20 64 72 6F 63 65 52 20 6E 61 65 4D     02BFC C.ANM:   .ASCII  \Mean Record Size \<0><0><0>
                                                  00 00 00 20 20 20 20 20   02C0A
                              20 20 20 65 7A 69 53 20 64 72 6F 63 65 52     02C10 C.ANN:   .ASCII  \Record Size      \<0><0><0>
                                                  00 00 00 20 64 6F 68 74   02C1E
                              64 6F 68 74 65 4D 20 64 61 6F 4C 2D 4D 4C     02C24 C.ANO:   .ASCII  \LM-Load Method \<0>
                                                                  00 20     02C32
                                  00 20 76 6E 6F 43 5F 74 73 61 46 20       02C34 C.ANP:   .ASCII  \ Fast_Conv \<0>
                                  00 20 6E 6F 43 5F 74 73 61 46 6F 4E       02C40 C.ANQ:   .ASCII  \NoFast_Con \<0>
                                  00 20 73 74 75 50 5F 53 4D 52 20 20       02C4C C.ANR:   .ASCII  \  RMS_Puts \<0>
                              61 6F 4C 20 6C 61 69 74 69 6E 49 2D 4C 49     02C58 C.ANS:   .ASCII  \IL-Initial Load \
                                                                  20 64     02C66
                              72 6F 63 65 52 20 64 65 64 64 41 2D 52 41     02C68 C.ANT:   .ASCII  \AR-Added Records\
                                                                  73 64     02C76
                                                              5C 1B 3B 27   02C78 C.ANU:   .ASCII  \';\<27><92>
                              6F 20 6C 6C 69 46 20 6C 61 69 74 69 6E 49     02C7C C.ANV:   .ASCII  \Initial Fill of 50% assumed\<0>
                              00 64 65 6D 75 73 73 61 20 25 30 35 20 66     02C8A
                                 00000000 00000000 00000000 00000102 00000000   02C98 C.ANW:   .LONG   0,^X102,0,0,0,0,0,0
                                                  00000000 00000000 00000000   02CAC
                                 00000000 00000000 00000001 45010004 00000000   02CB8 C.ANX:   .LONG   0,^X45010004,1,0,0,0,0,0
                                                  00000000 00000000 00000000   02CCC
                                 00000000 00000000 00000000 00000000 00070000   02CD8 C.ANY:   .LONG   ^X70000,0,0,0,0,0,0,0
                                                  00000000 00000000 00000000   02CEC
                              75 20 72 6F 74 63 61 46 20 6C 6C 69 46 20     02CF8 C.ANZ:   .ASCII  \ Fill Factor used is 100% when Initial L\-
                              68 77 20 25 30 30 31 20 73 69 20 64 65 73     02D06                  \oad is zero. \<0><0><0>
                              61 6F 4C 20 6C 61 69 74 69 6E 49 20 6E 65     02D14
                              00 00 00 20 2E 6F 72 65 7A 20 73 69 20 64     02D22
                                 00000000 00000000 00000000 00000000 00006000   02D30 C.AOA:   .LONG   ^X6000,0,0,0,0,0,0,0
                                                  00000000 00000000 00000000   02D44
                                 00000000 00000000 00000000 00000000 00071000   02D50 C.AOB:   .LONG   ^X71000,0,0,0,0,0,0,0
                                                  00000000 00000000 00000000   02D64
                                 00000000 00000000 00000000 00000000 00000067   02D70 C.AOC:   .LONG   ^X67,0,0,0,0,0,0,0
                                                  00000000 00000000 00000000   02D84
                                    00 5C 1B 3B 29 45 28 53 3B 70 50 1B       02D90 C.AOD:   .ASCII  <27>\Pp;S(E);\<27><92><0>
                                                              00 00 09 09     02D9C C.AOE:   .ASCII  <9><9><0><0>
                              00 53 53 45 43 43 41 20 6C 61 67 65 4C 20     02DA0 C.AOF:   .ASCII  \ Legal ACCESS\<0><0><0>
                                                                  00 00     02DAE
                              2F 73 65 79 09 09 4F 49 5F 4B 43 4F 4C 42     02DB0 C.AOG:   .ASCII  \BLOCK_IO\<9><9>\yes/no\
```

```
                                                                  6F 6E     02DBE
6E 2F 73 65 79 09 09 09 45 54 45 4C 45 44     02DC0 C.AOH:  .ASCII  \DELETE\<9><9><9>\yes/no\<0>
                                              00 6F     02DCE
   6F 6E 2F 73 65 79 09 09 09 54 45 47     02DD0 C.AOI:  .ASCII  \GET\<9><9><9>\yes/no\
   6F 6E 2F 73 65 79 09 09 09 54 55 50     02DDC C.AOJ:  .ASCII  \PUT\<9><9><9>\yes/no\
73 65 79 09 09 4F 49 5F 44 52 4F 43 45 52     02DE8 C.AOK:  .ASCII  \RECORD_IO\<9><9>\yes/no\<0><0><0>
                        00 00 00 6F 6E 2F     02DF6
2F 73 65 79 09 09 45 54 41 43 4E 55 52 54     02DFC C.AOL:  .ASCII  \TRUNCATE\<9><9>\yes/no\
                                              6F 6E     02E0A
6E 2F 73 65 79 09 09 09 45 54 41 44 50 55     02E0C C.AOM:  .ASCII  \UPDATE\<9><9><9>\yes/no\<0>
                                              00 6F     02E1A
                                 00 00 09 09     02E1C C.AON:  .ASCII  <9><9><0><0>
53 45 43 43 41 20 74 6E 65 72 72 75 43 20     02E20 C.AOO:  .ASCII  \ Current ACCESS\<0>
                                              00 53     02E2E
41 20 53 53 45 43 43 41 20 72 65 74 6E 45     02E30 C.AOP:  .ASCII  \Enter ACCESS Attribute\<9><9>\(Keyword)\-
79 65 4B 28 09 09 65 74 75 62 69 72 74 74     02E3E                 <0><0><0>
                  00 00 00 29 64 72 6F 77     02E4C
                                 00 5D 2D 5B     02E54 C.AOQ:  .ASCII  \[-]\<0>
                                 00 20 3A 09     02E58 C.AOR:  .ASCII  <9>\: \<0>
                                 00 00 09 09     02E5C C.AOS:  .ASCII  <9><9><0><0>
   20 41 45 52 41 20 6C 61 67 65 4C 20     02E60 C.AOT:  .ASCII  \ Legal AREA \
75 6E 09 09 4E 4F 49 54 41 43 4F 4C 4C 41     02E6C C.AOU:  .ASCII  \ALLOCATION\<9><9>\number\<0><0>
                        00 00 72 65 62 6D     02E7A
49 54 4E 4F 43 5F 59 52 54 5F 54 53 45 42     02E80 C.AOV:  .ASCII  \BEST_TRY_CONTIGUOUS\<9>\yes/no\<0><0>
00 00 6F 6E 2F 73 65 79 09 53 55 4F 55 47     02E8E
6E 09 09 45 5A 49 53 5F 54 45 4B 43 55 42     02E9C C.AOW:  .ASCII  \BUCKET_SIZE\<9><9>\number\<0>
                        00 72 65 62 6D 75     02EAA
65 79 09 09 53 55 4F 55 47 49 54 4E 4F 43     02EB0 C.AOX:  .ASCII  \CONTIGUOUS\<9><9>\yes/no\<0><0>
                        00 00 6F 6E 2F 73     02EBE
4E 4F 49 54 49 53 4F 50 5F 54 43 41 58 45     02EC4 C.AOY:  .ASCII  \EXACT_POSITIONING\<9>\yes/no\
            6F 6E 2F 73 65 79 09 47 4E 49     02ED2
6D 75 6E 09 09 4E 4F 49 53 4E 45 54 58 45     02EDC C.AOZ:  .ASCII  \EXTENSION\<9><9>\number\<0><0><0>
                        00 00 00 72 65 62     02EEA
75 71 20 20 20 20 4E 4F 49 54 49 53 4F 50     02EF0 C.APA:  .ASCII  \POSITION    qualifier\<9>\number\
72 65 62 6D 75 6E 09 72 65 69 66 69 6C 61     02EFE
65 62 6D 75 6E 09 09 09 45 4D 55 4C 4F 56     02F0C C.APB:  .ASCII  \VOLUME\<9><9><9>\number\<0>
                                              00 72     02F1A
                                 00 00 09 09     02F1C C.APC:  .ASCII  <9><9><0><0>
20 41 45 52 41 20 74 6E 65 72 72 75 43 20     02F20 C.APD:  .ASCII  \ Current AREA \<0><0>
                                              00 00     02F2E
   00 20 41 45 52 41 20 72 65 74 6E 45     02F30 C.APE:  .ASCII  \Enter AREA \<0>
4B 28 09 09 65 74 75 62 69 72 74 74 41 20     02F3C C.APF:  .ASCII  \ Attribute\<9><9>\(Keyword)\<0><0><0>
            00 00 00 29 64 72 6F 77 79 65     02F4A
                                 00 5D 2D 5B     02F54 C.APG:  .ASCII  \[-]\<0>
                                 00 20 3A 09     02F58 C.APH:  .ASCII  <9>\: \<0>
                                 00 00 09 09     02F5C C.API:  .ASCII  <9><9><0><0>
54 43 45 4E 4E 4F 43 20 6C 61 67 65 4C 20     02F60 C.APJ:  .ASCII  \ Legal CONNECT\<0><0>
                                              00 00     02F6E
09 09 53 55 4F 4E 4F 52 48 43 4E 59 53 41     02F70 C.APK:  .ASCII  \ASYNCHRONOUS\<9><9>\yes/no\<9>\NOLOCK\-
09 4B 43 4F 4C 4F 4E 09 6F 6E 2F 73 65 79     02F7E                 <9><9><9>\yes/no\
                  6F 6E 2F 73 65 79 09 09     02F8C
2F 73 65 79 09 09 4F 49 5F 4B 43 4F 4C 42     02F94 C.APL:  .ASCII  \BLOCK_IO\<9><9>\yes/no\<9>\NONEXISTENT_R\-
54 4E 45 54 53 49 58 45 4E 4F 4E 09 6F 6E     02FA2                 \ECORD\<9>\yes/no\<0><0>
6F 6E 2F 73 65 79 09 44 52 4F 43 45 52 5F     02FB0
                                              00 00     02FBE
6E 09 09 45 44 4F 43 5F 54 45 4B 43 55 42     02FC0 C.APM:  .ASCII  \BUCKET_CODE\<9><9>\number\<9>\READ_AHEAD\-
45 48 41 5F 44 41 45 52 09 72 65 62 6D 75     02FCE                 <9><9>\yes/no\<0><0>
```

```
                00 00 6F 6E 2F 73 65 79 09 09 44 41      02FDC
62 6D 75 6E 09 09 09 54 58 45 54 4E 4F 43      02FE8 C.APN:   .ASCII   \CONTEXT\<9><9><9>\number\<9>\READ_REGARD\-
44 52 41 47 45 52 5F 44 41 45 52 09 72 65      02FF6                  \LESS\<9><9>\yes/no\
         6F 6E 2F 73 65 79 09 09 53 53 45 4C   03004
79 09 09 45 4C 49 46 5F 46 4F 5F 44 4E 45      03010 C.APO:   .ASCII   \END_OF_FILE\<9><9>\yes/no\<9>\TIMEOUT_EN\-
5F 54 55 4F 45 4D 49 54 09 6F 6E 2F 73 65      0301E                  \ABLE\<9><9>\yes/no\<0><0>
6F 6E 2F 73 65 79 09 09 45 4C 42 41 4E 45      0302C
                                  00 00        0303A
79 09 09 45 54 45 4C 45 44 5F 54 53 41 46      0303C C.APP:   .ASCII   \FAST_DELETE\<9><9>\yes/no\<9>\TIMEOUT_PE\-
5F 54 55 4F 45 4D 49 54 09 6F 6E 2F 73 65      0304A                  \RIOD\<9><9>\number\<0><0>
72 65 62 6D 75 6E 09 09 44 4F 49 52 45 50      03058
                                  00 00        03066
09 09 53 54 45 4B 43 55 42 5F 4C 4C 49 46      03068 C.APQ:   .ASCII   \FILL_BUCKETS\<9><9>\yes/no\<9>\TRUNCATE_\-
54 41 43 4E 55 52 54 09 6F 6E 2F 73 65 79      03076                  \ON_POT\<9><9>\yes/no\
2F 73 65 79 09 09 54 55 50 5F 4E 4F 5F 45      03084
                                  6F 6E        03092
51 45 5F 52 45 54 41 45 52 47 5F 59 45 4B      03094 C.APR:   .ASCII   \KEY_GREATER_EQUAL\<9>\yes/no\<9>\TT_CANC\-
5F 54 54 09 6F 6E 2F 73 65 79 09 4C 41 55      030A2                  \EL_CONTROL_O\<9>\yes/no\<0>
4C 4F 52 54 4E 4F 43 5F 4C 45 43 4E 41 43      030B0
                  00 6F 6E 2F 73 65 79 09 4F 5F 030BE
48 54 5F 52 45 54 41 45 52 47 5F 59 45 4B      030C8 C.APS:   .ASCII   \KEY_GREATER_THAN\<9>\yes/no\<9>\TT_PROMP\-
50 5F 54 54 09 6F 6E 2F 73 65 79 09 4E 41      030D6                  \T\<9><9>\yes/no\<0><0><0>
00 6F 6E 2F 73 65 79 09 09 54 50 4D 4F 52      030E4
                                  00 00        030F2
73 65 79 09 09 54 49 4D 49 4C 5F 59 45 4B      030F4 C.APT:   .ASCII   \KEY_LIMIT\<9><9>\yes/no\<9>\TT_PURGE_TYP\-
54 5F 45 47 52 55 50 5F 54 54 09 6F 6E 2F      03102                  \E_AREAD\<9>\yes/no\
2F 73 65 79 09 44 41 45 48 41 5F 45 50 59      03110
                                  6F 6E        0311E
4E 45 52 45 46 45 52 5F 46 4F 5F 59 45 4B      03120 C.APU:   .ASCII   \KEY_OF_REFERENCE\<9>\number\<9>\TT_READ_\-
52 5F 54 54 09 72 65 62 6D 75 6E 09 45 43      0312E                  \NOECHO\<9><9>\yes/no\<0><0>
65 79 09 09 4F 48 43 45 4F 4E 5F 44 41 45      0313C
                        00 00 6F 6E 2F 73      0314A
79 09 09 45 44 4F 4D 5F 45 54 41 43 4F 4C      03150 C.APV:   .ASCII   \LOCATE_MODE\<9><9>\yes/no\<9>\TT_READ_NO\-
5F 44 41 45 52 5F 54 54 09 6F 6E 2F 73 65      0315E                  \FILTER\<9>\yes/no\<0>
6E 2F 73 65 79 09 52 45 54 4C 49 46 4F 4E      0316C
                                  00 6F        0317A
09 09 44 41 45 52 5F 4E 4F 5F 4B 43 4F 4C      0317C C.APW:   .ASCII   \LOCK_ON_READ\<9><9>\yes/no\<9>\TT_UPCASE\-
41 43 50 55 5F 54 54 09 6F 6E 2F 73 65 79      0318A                  \_INPUT\<9><9>\yes/no\
2F 73 65 79 09 09 54 55 50 4E 49 5F 45 53      03198
                                  6F 6E        031A6
09 45 54 49 52 57 5F 4E 4F 5F 4B 43 4F 4C      031A8 C.APX:   .ASCII   \LOCK_ON_WRITE\<9><9>\yes/no\<9>\UPDATE_I\-
45 54 41 44 50 55 09 6F 6E 2F 73 65 79 09      031B6                  \F\<9><9>\yes/no\<0>
         00 6F 6E 2F 73 65 79 09 09 46 49 5F   031C4
49 4B 43 4F 4C 4E 55 5F 4C 41 55 4E 41 4D      031D0 C.APY:   .ASCII   \MANUAL_UNLOCKING\<9>\yes/no\<9>\WAIT_FOR\-
54 49 41 57 09 6F 6E 2F 73 65 79 09 47 4E      031DE                  \_RECORD\<9><9>\yes/no\<0>
79 09 09 44 52 4F 43 45 52 5F 52 4F 46 5F      031EC
                        00 6F 6E 2F 73 65      031FA
55 4F 43 5F 4B 43 4F 4C 42 49 54 4C 55 4D      03200 C.APZ:   .ASCII   \MULTIBLOCK_COUNT\<9>\number\<9>\WRITE_BE\-
54 49 52 57 09 72 65 62 6D 75 6E 09 54 4E      0320E                  \HIND\<9><9>\yes/no\
2F 73 65 79 09 09 44 4E 49 48 45 42 5F 45      0321C
                                  6F 6E        0322A
4F 43 5F 52 45 46 46 55 42 49 54 4C 55 4D      0322C C.AQA:   .ASCII   \MULTIBUFFER_COUNT\<9>\number\
         72 65 62 6D 75 6E 09 54 4E 55         0323A
                        00 00 09 09            03244 C.AQB:   .ASCII   <9><9><0><0>
45 4E 4E 4F 43 20 74 6E 65 72 72 75 43 20      03248 C.AQC:   .ASCII   \ Current CONNECT\
                                  54 43        03256
20 54 43 45 4E 4E 4F 43 20 72 65 74 6E 45      03258 C.AQD:   .ASCII   \Enter CONNECT Attribute\<9><9>\(Keyword)\-
```

```
65 4B 28 09 09 65 74 75 62 69 72 74 74 41
                  00 00 29 64 72 6F 77 79
                                 00 5D 2D 5B
                                 00 20 3A 09
                                 00 00 09 09
      00 45 54 41 44 20 6C 61 67 65 4C 20
6E 69 72 74 73 09 09 09 50 55 4B 43 41 42
                                    00 67
69 72 74 73 09 09 4E 4F 49 54 41 45 52 43
                                    67 6E
74 73 09 09 4E 4F 49 54 41 52 49 50 58 45
                  00 00 67 6E 69 72
69 72 74 73 09 09 4E 4F 49 53 49 56 45 52
                                    67 6E
                                 00 00 09 09
00 45 54 41 44 20 74 6E 65 72 72 75 43 20
                                    00 00
74 74 41 20 45 54 41 44 20 72 65 74 6E 45
6F 77 79 65 4B 28 09 09 65 74 75 62 69 72
                              00 29 64 72
                              00 5D 2D 5B
                              00 20 3A 09
                              00 00 09 09
      00 45 4C 49 46 20 6C 61 67 65 4C 20
75 6E 09 09 4E 4F 49 54 41 43 4F 4C 4C 41
43 45 54 4F 52 50 5F 54 4D 09 72 65 62 6D
6D 75 6E 2F 72 61 68 63 09 09 4E 4F 49 54
                                    00 00
49 54 4E 4F 43 5F 59 52 54 5F 54 53 45 42
4E 09 6F 6E 2F 73 65 79 09 53 55 4F 55 47
      67 6E 69 72 74 73 09 09 09 45 4D 41
6E 09 09 45 5A 49 53 5F 54 45 4B 43 55 42
50 55 4B 43 41 42 4F 4E 09 72 65 62 6D 75
                  6F 6E 2F 73 65 79 09 09
09 09 45 5A 49 53 5F 52 45 54 53 55 4C 43
4C 49 46 5F 4E 4F 4E 09 72 65 62 6D 75 6E
79 09 44 45 52 55 54 43 55 52 54 53 5F 45
                              00 6F 6E 2F 73 65
62 6D 75 6E 09 09 09 54 58 45 54 4E 4F 43
4F 49 54 41 5A 49 4E 41 47 52 4F 09 72 65
      00 00 64 72 6F 77 79 65 6B 09 09 4E
65 79 09 09 53 55 4F 55 47 49 54 4E 4F 43
49 46 5F 54 55 50 54 55 4F 09 6F 6E 2F 73
6E 2F 73 65 79 09 45 53 52 41 50 5F 45 4C
                                 00 6F
73 65 79 09 09 46 49 5F 45 54 41 45 52 43
69 75 09 09 09 52 45 4E 57 4F 09 6F 6E 2F
                              00 00 00 63
09 09 45 4D 41 4E 5F 54 4C 55 41 46 45 44
4F 5F 54 4E 49 52 50 09 67 6E 69 72 74 73
6E 2F 73 65 79 09 09 45 53 4F 4C 43 5F 4E
                                 00 6F
45 54 49 52 57 5F 44 45 52 52 45 46 45 44
45 54 4F 52 50 09 6F 6E 2F 73 65 79 09 09
00 6F 6E 2F 73 65 79 09 09 4E 4F 49 54 43
                                 00 00
53 4F 4C 43 5F 4E 4F 5F 45 54 45 4C 45 44
```

```
03266                   <0><0>
03274
0327C C.AQE:  .ASCII    \[-]\<0>
03280 C.AQF:  .ASCII    <9>\: \<0>
03284 C.AQG:  .ASCII    <9><9><0><0>
03288 C.AQH:  .ASCII    \ Legal DATE\<0>
03294 C.AQI:  .ASCII    \BACKUP\<9><9><9>\string\<0>
032A2
032A4 C.AQJ:  .ASCII    \CREATION\<9><9>\string\
032B2
032B4 C.AQK:  .ASCII    \EXPIRATION\<9><9>\string\<0><0>
032C2
032C8 C.AQL:  .ASCII    \REVISION\<9><9>\string\
032D6
032D8 C.AQM:  .ASCII    <9><9><0><0>
032DC C.AQN:  .ASCII    \ Current DATE\<0><0><0>
032EA
032EC C.AQO:  .ASCII    \Enter DATE Attribute\<9><9>\(Keyword)\<0>
032FA
03308
0330C C.AQP:  .ASCII    \[-]\<0>
03310 C.AQQ:  .ASCII    <9>\: \<0>
03314 C.AQR:  .ASCII    <9><9><0><0>
03318 C.AQS:  .ASCII    \ Legal FILE\<0>
03324 C.AQT:  .ASCII    \ALLOCATION\<9><9>\number\<9>\MT_PROTECTI\-
03332                   \ON\<9><9>\char/num\<0><0>
03340
0334E
03350 C.AQU:  .ASCII    \BEST_TRY_CONTIGUOUS\<9>\yes/no\<9>\NAME\-
0335E                   <9><95<9>\string\
0336C
03378 C.AQV:  .ASCII    \BUCKET_SIZE\<9><9>\number\<9>\NOBACKUP\-
03386                   <9><9>\yes/no\
03394
0339C C.AQW:  .ASCII    \CLUSTER_SIZE\<9><9>\number\<9>\NON_FILE_\-
033AA                   \STRUCTURED\<9>\yes/no\<0>
033B8
033C6
033CC C.AQX:  .ASCII    \CONTEXT\<9><9><9>\number\<9>\ORGANIZATIO\-
033DA                   \N\<9><9>\keyword\<0><0>
033E8
033F4 C.AQY:  .ASCII    \CONTIGUOUS\<9><9>\yes/no\<9>\OUTPUT_FILE\-
03402                   \_PARSE\<9>\yes/no\<0>
03410
0341E
03420 C.AQZ:  .ASCII    \CREATE_IF\<9><9>\yes/no\<9>\OWNER\<9>-
0342E                   <9><9>\uic\<0><0><0>
0343C
03440 C.ARA:  .ASCII    \DEFAULT_NAME\<9><9>\string\<9>\PRINT_ON_\-
0344E                   \CLOSE\<9><9>\yes/no\<0>
0345C
0346A
0346C C.ARB:  .ASCII    \DEFERRED_WRITE\<9><9>\yes/no\<9>\PROTECT\-
0347A                   \ION\<9><9>\yes/no\<0><0><0>
03488
03496
03498 C.ARC:  .ASCII    \DELETE_ON_CLOSE\<9><9>\yes/no\<9>\READ_C\-
```

```
44 41 45 52 09 6F 6E 2F 73 65 79 09 09 45    034A6                   \HECK\<9><9>\yes/no\<0><0>
6F 6E 2F 73 65 79 09 09 4B 43 45 48 43 5F    034B4
                                  00 00      034C2
52 54 4E 45 5F 59 52 4F 54 43 45 52 49 44    034C4 C.ARD:  .ASCII    \DIRECTORY_ENTRY\<9><9>\yes/no\<9>\REVISI\-
49 56 45 52 09 6F 6E 2F 73 65 79 09 09 59    034D2                   \ON\<9><9>\number\
      72 65 62 6D 75 6E 09 09 4E 4F 49 53    034E0
6D 75 6E 09 09 4E 4F 49 53 4E 45 54 58 45    034EC C.ARE:  .ASCII    \EXTENSION\<9><9>\number\<9>\SEQUENTIAL_O\-
4C 41 49 54 4E 45 55 51 45 53 09 72 65 62    034FA                   \NLY\<9><9>\yes/no\<0><0><0>
00 6F 6E 2F 73 65 79 09 09 59 4C 4E 4F 5F    03508
                                  00 00      03516
5F 52 45 46 46 55 42 5F 4C 41 42 4F 4C 47    03518 C.ARF:  .ASCII    \GLOBAL_BUFFER_COUNT\<9>\number\<9>\SUBMI\-
53 09 72 65 62 6D 75 6E 09 54 4E 55 4F 43    03526                   \T_ON_CLOSE\<9><9>\yes/no\<0><0>
45 53 4F 4C 43 5F 4E 4F 5F 54 49 4D 42 55    03534
         00 00 6F 6E 2F 73 65 79 09 09       03542
4D 55 4E 5F 44 52 4F 43 45 52 5F 58 41 4D    0354C C.ARG:  .ASCII    \MAX_RECORD_NUMBER\<9>\number\<9>\SUPERSE\-
50 55 53 09 72 65 62 6D 75 6E 09 52 45 42    0355A                   \DE\<9><9>\yes/no\<0><0>
6F 6E 2F 73 65 79 09 09 45 44 45 53 52 45    03568
                                  00 00      03576
49 53 52 45 56 5F 45 5A 49 4D 49 58 41 4D    03578 C.ARH:  .ASCII    \MAXIMIZE_VERSION\<9>\yes/no\<9>\TEMPORAR\-
50 4D 45 54 09 6F 6E 2F 73 65 79 09 4E 4F    03586                   \Y\<9><9>\yes/no\<0><0><0>
00 6F 6E 2F 73 65 79 09 09 59 52 41 52 4F    03594
                                  00 00      035A2
09 45 5A 49 53 5F 4B 43 4F 4C 42 5F 54 4D    035A4 C.ARI:  .ASCII    \MT_BLOCK_SIZE\<9><9>\number\<9>\TRUNCATE\-
41 43 4E 55 52 54 09 72 65 62 6D 75 6E 09    035B2                   \_ON_CLOSE\<9>\yes/no\<0><0>
65 79 09 45 53 4F 4C 43 5F 4E 4F 5F 45 54    035C0
                  00 00 6F 6E 2F 73          035CE
4E 49 57 45 52 5F 45 53 4F 4C 43 5F 54 4D    035D4 C.ARJ:  .ASCII    \MT_CLOSE_REWIND\<9><9>\yes/no\<9>\USER_F\-
52 45 53 55 09 6F 6E 2F 73 65 79 09 09 44    035E2                   \ILE_OPEN\<9><9>\yes/no\<0><0>
65 79 09 09 4E 45 50 4F 5F 45 4C 49 46 5F    035F0
                  00 00 6F 6E 2F 73          035FE
53 4F 50 5F 54 4E 45 52 52 55 43 5F 54 4D    03604 C.ARK:  .ASCII    \MT_CURRENT_POSITION\<9>\yes/no\<9>\WINDO\-
57 09 6F 6E 2F 73 65 79 09 4E 4F 49 54 49    03612                   \W_SIZE\<9><9>\number\<0><0>
75 6E 09 09 45 5A 49 53 5F 57 4F 44 4E 49    03620
                  00 00 72 65 62 6D          0362E
65 79 09 09 46 4F 45 5F 54 4F 4E 5F 54 4D    03634 C.ARL:  .ASCII    \MT_NOT_EOF\<9><9>\yes/no\<9>\WRITE_CHECK\-
45 48 43 5F 45 54 49 52 57 09 6F 6E 2F 73    03642                   <9><9>\yes/no\
            6F 6E 2F 73 65 79 09 09 4B 43    03650
                                  09 09      0365A C.ARM:  .ASCII    <9><9>
00 45 4C 49 46 20 74 6E 65 72 72 75 43 20    0365C C.ARN:  .ASCII    \ Current FILE\<0><0><0>
                                  00 00      0366A
74 74 41 20 45 4C 49 46 20 72 65 74 6E 45    0366C C.ARO:  .ASCII    \Enter FILE Attribute\<9><9>\(Keyword)\<0>
6F 77 79 65 4B 28 09 09 65 74 75 62 69 72    0367A
                        00 29 64 72          03688
                        00 5D 2D 5B          0368C C.ARP:  .ASCII    \[-]\<0>
                        00 20 3A 09          03690 C.ARQ:  .ASCII    <9>\: \<0>
                        00 00 09 09          03694 C.ARR:  .ASCII    <9><9><0><0>
4C 41 4E 52 55 4F 4A 20 6C 61 67 65 4C 20    03698 C.ARS:  .ASCII    \ Legal JOURNAL\<0><0>
                                  00 00      036A6
72 09 09 45 47 41 4D 49 5F 52 45 54 46 41    036A8 C.ART:  .ASCII    \AFTER_IMAGE\<9><9>\yes/no\<0>
                  00 6F 6E 2F 73 65          036B6
74 73 09 09 45 4D 41 4E 5F 52 45 54 46 41    036BC C.ARU:  .ASCII    \AFTER_NAME\<9><9>\string\<0><0>
                  00 00 67 6E 69 72          036CA
79 09 09 4C 49 41 52 54 5F 54 49 44 55 41    036D0 C.ARV:  .ASCII    \AUDIT_TRAIL\<9><9>\yes/no\<0>
                  00 6F 6E 2F 73 65          036DE
74 73 09 09 45 4D 41 4E 5F 54 49 44 55 41    036E4 C.ARW:  .ASCII    \AUDIT_NAME\<9><9>\string\<0><0>
                  00 00 67 6E 69 72          036F2
09 09 45 47 41 4D 49 5F 45 52 4F 46 45 42    036F8 C.ARX:  .ASCII    \BEFORE_IMAGE\<9><9>\yes/no\
```

```
                                                 6F 6E 2F 73 65 79       03706
 73 09 09 45 4D 41 4E 5F 45 52 4F 46 45 42       0370C C.ARY:  .ASCII  \BEFORE_NAME\<9><9>\string\<0>
                         00 67 6E 69 72 74       0371A
 09 54 49 4E 55 5F 59 52 45 56 4F 43 45 52       03720 C.ARZ:  .ASCII  \RECOVERY_UNIT\<9><9>\keyword\
                      64 72 6F 77 79 65 6B 09    0372E
                                     09 09       03736 C.ASA:  .ASCII  <9><9>
 4E 52 55 4F 4A 20 74 6E 65 72 72 75 43 20       03738 C.ASB:  .ASCII  \ Current JOURNAL\
                                  4C 41          03746
 20 4C 41 4E 52 55 4F 4A 20 72 65 74 6E 45       03748 C.ASC:  .ASCII  \Enter JOURNAL Attribute\<9><9>\(Keyword)\-
 65 4B 28 09 09 65 74 75 62 69 72 74 74 41       03756                 <0><0>
                   00 00 29 64 72 6F 77 79       03764
                                  00 5D 2D 5B      0376C C.ASD:  .ASCII  \[-]\<0>
                                  00 20 3A 09      03770 C.ASE:  .ASCII  <9>\: \<0>
                                  00 00 09 09      03774 C.ASF:  .ASCII  <9><9><0><0>
          00 00 59 45 4B 20 6C 61 67 65 4C 20     03778 C.ASG:  .ASCII  \ Legal KEY\<0><0>
 2F 73 65 79 09 09 09 53 45 47 4E 41 48 43       03784 C.ASH:  .ASCII  \CHANGES\<9><9><9>\yes/no\<9>\LEVEL1_INDE\-
 45 44 4E 49 5F 31 4C 45 56 45 4C 09 6F 6E       03792                 \X_AREA\<9>\number\<0><0><0>
 00 72 65 62 6D 75 6E 09 41 45 52 41 5F 58       037A0
                                  00 00          037AE
 6D 75 6E 09 09 41 45 52 41 5F 41 54 41 44       037B0 C.ASI:  .ASCII  \DATA_AREA\<9><9>\number\<9>\NAME\<9>-
 72 74 73 09 09 09 45 4D 41 4E 09 72 65 62       037BE                 <9><95\string\<0>
                                  00 67 6E 69    037CC
 6D 75 6E 09 09 4C 4C 49 46 5F 41 54 41 44       037D0 C.ASJ:  .ASCII  \DATA_FILL\<9><9>\number\<9>\NULL_KEY\-
 09 09 59 45 4B 5F 4C 4C 55 4E 09 72 65 62       037DE                 <9><95\yes/no\<0><0>
                   00 00 6F 6E 2F 73 65 79       037EC
 52 50 4D 4F 43 5F 59 45 4B 5F 41 54 41 44       037F4 C.ASK:  .ASCII  \DATA_KEY_COMPRESSION\<9>\yes/no\<9>\NULL\-
 09 6F 6E 2F 73 65 79 09 4E 4F 49 53 53 45       03802                 \_VALUE\<9><9>\char/num\
 68 63 09 09 45 55 4C 41 56 5F 4C 4C 55 4E       03810
                                  6D 75 6E 2F    0381E
 4F 43 5F 44 52 4F 43 45 52 5F 41 54 41 44       03824 C.ASL:  .ASCII  \DATA_RECORD_COMPRESSION\<9>\yes/no\<9>\P\-
 2F 73 65 79 09 4E 4F 49 53 53 45 52 50 4D       03832                 \OSITION\<9><9>\number\<0>
 6E 09 09 4E 4F 49 54 49 53 4F 50 09 6F 6E       03840
                                  00 72 65 62    0384E
 65 79 09 09 53 45 54 41 43 49 4C 50 55 44       03854 C.ASM:  .ASCII  \DUPLICATES\<9><9>\yes/no\<9>\PROLOG\-
 09 09 09 47 4F 4C 4F 52 50 09 6F 6E 2F 73       03862                 <9><9><9>\number\<0><0>
                   00 00 72 65 62 6D 75 6E       03870
 75 6E 09 09 41 45 52 41 5F 58 45 44 4E 49       03878 C.ASN:  .ASCII  \INDEX_AREA\<9><9>\number\<9>\TYPE\<9>-
 65 6B C9 09 09 45 50 59 54 09 72 65 62 6D       03886                 <9><9>\keyword\<0><0><0>
                   00 00 00 64 72 6F 77 79       03894
 53 53 45 52 50 4D 4F 43 5F 58 45 44 4E 49       0389C C.ASO:  .ASCII  \INDEX_COMPRESSION\<9>\yes/no\<9>\SEGn_LE\-
 47 45 53 09 6F 6E 2F 73 65 79 09 4E 4F 49       038AA                 \NGTH\<9><9>\number\
 62 6D 75 6E 09 09 48 54 47 4E 45 4C 5F 6E       038B8
                                  72 65          038C6
 75 6E 09 09 4C 4C 49 46 5F 58 45 44 4E 49       038C8 C.ASP:  .ASCII  \INDEX_FILL\<9><9>\number\<9>\SEGn_POSITI\-
 49 53 4F 50 5F 6E 47 45 53 09 72 65 62 6D       038D6                 \ON\<95<9>\number\
          72 65 62 6D 75 6E 09 09 4E 4F 49 54    038E4
 65 62 6D 75 6E 09 09 09 48 54 47 4E 45 4C       038F0 C.ASQ:  .ASCII  \LENGTH\<9><9><9>\number\<0>
                                  00 72          038FE
                                  00 00 09 09    03900 C.ASR:  .ASCII  <9><9><0><0>
          59 45 4B 20 74 6E 65 72 72 75 43 20    03904 C.ASS:  .ASCII  \ Current KEY\
          00 00 00 59 45 4B 20 72 65 74 6E 45    03910 C.AST:  .ASCII  \Enter KEY\<0><0><0>
 4B 28 09 09 65 74 75 62 69 72 74 74 41 20       0391C C.ASU:  .ASCII  \ Attribute\<9><9>\(Keyword)\<0><0><0>
             00 00 00 29 64 72 6F 77 79 65       0392A
                                  00 5D 2D 5B    03934 C.ASV:  .ASCII  \[-]\<0>
                                  00 20 3A 09    03938 C.ASW:  .ASCII  <9>\: \<0>
                                  00 00 09 09    0393C C.ASX:  .ASCII  <9><9><0><0>
 00 44 52 4F 43 45 52 20 6C 61 67 65 4C 20       03940 C.ASY:  .ASCII  \ Legal RECORD\<0><0><0>
```

```
                                                                  00 00     0394E
65 79 09 09 4E 41 50 53 5F 4B 43 4F 4C 42     03950 C.ASZ:  .ASCII  \BLOCK_SPAN\<9><9>\yes/no\<0><0>
                        00 00 6F 6E 2F 73     0395E
52 54 4E 4F 43 5F 45 47 41 49 52 52 41 43     03964 C.ATA:  .ASCII  \CARRIAGE_CONTROL\<9>\keyword\
            64 72 6F 77 79 65 6B 09 4C 4F     03972
5F 44 4C 45 49 46 5F 4C 4F 52 54 4E 4F 43     0397C C.ATB:  .ASCII  \CONTROL_FIELD_SIZE\<9>\number\<0><0><0>
00 00 00 72 65 62 6D 75 6E 09 45 5A 49 53     0398A
6F 77 79 65 6B 09 09 09 54 41 4D 52 4F 46     03998 C.ATC:  .ASCII  \FORMAT\<9><9><9>\keyword\
                                  64 72       039A6
00 72 65 62 6D 75 6E 09 09 09 45 5A 49 53     039A8 C.ATD:  .ASCII  \SIZE\<9><9><9>\number\<0>
                                  09 09       039B6 C.ATE:  .ASCII  <9><9>
52 4F 43 45 52 20 74 6E 65 72 72 75 43 20     039B8 C.ATF:  .ASCII  \ Current RECORD\<0>
                                  00 44       039C6
41 20 44 52 4F 43 45 52 20 72 65 74 6E 45     039C8 C.ATG:  .ASCII  \Enter RECORD Attribute\<9><9>\(Keyword)\-
79 65 4B 28 09 09 65 74 75 62 69 72 74 74     039D6                 <0><0><0>
                  00 00 00 29 64 72 6F 77     039E4
                              00 5D 2D 5B     039EC C.ATH:  .ASCII  \[-]\<0>
                              00 20 3A 09     039F0 C.ATI:  .ASCII  <9>\: \<0>
                              00 00 09 09     039F4 C.ATJ:  .ASCII  <9><9><0><0>
47 4E 49 52 41 48 53 20 6C 61 67 65 4C 20     039F8 C.ATK:  .ASCII  \ Legal SHARING\<0><0>
                                  00 00       03A06
6E 2F 73 65 79 09 09 09 45 54 45 4C 45 44     03A08 C.ATL:  .ASCII  \DELETE\<9><9><9>\yes/no\<0>
                                  00 6F       03A16
      6F 6E 2F 73 65 79 09 09 09 54 45 47     03A18 C.ATM:  .ASCII  \GET\<9><9><9>\yes/no\
79 09 09 4D 41 45 52 54 53 49 54 4C 55 4D     03A24 C.ATN:  .ASCII  \MULTISTREAM\<9><9>\yes/no\<0>
                        00 6F 6E 2F 73 65     03A32
2F 73 65 79 09 09 54 49 42 49 48 4F 52 50     03A38 C.ATO:  .ASCII  \PROHIBIT\<9><9>\yes/no\
                                  6F 6E       03A46
      6F 6E 2F 73 65 79 09 09 09 54 55 50     03A48 C.ATP:  .ASCII  \PUT\<9><9><9>\yes/no\
6E 2F 73 65 79 09 09 09 45 54 41 44 50 55     03A54 C.ATQ:  .ASCII  \UPDATE\<9><9><9>\yes/no\<0>
                                  00 6F       03A62
4B 43 4F 4C 52 45 54 4E 49 5F 52 45 53 55     03A64 C.ATR:  .ASCII  \USER_INTERLOCK\<9><9>\yes/no\
                  6F 6E 2F 73 65 79 09 09     03A72
                                  09 09       03A7A C.ATS:  .ASCII  <9><9>
49 52 41 48 53 20 74 6E 65 72 72 75 43 20     03A7C C.ATT:  .ASCII  \ Current SHARING\
                                  47 4E       03A8A
20 47 4E 49 52 41 48 53 20 72 65 74 6E 45     03A8C C.ATU:  .ASCII  \Enter SHARING Attribute\<9><9>\(Keyword)\-
65 4B 28 09 09 65 74 75 62 69 72 74 74 41     03A9A                 <0><0>
                  00 00 29 64 72 6F 77 79     03AA8
                              00 5D 2D 5B     03AB0 C.ATV:  .ASCII  \[-]\<0>
                              00 20 3A 09     03AB4 C.ATW:  .ASCII  <9>\: \<0>
                              00 00 09 09     03AB8 C.ATX:  .ASCII  <9><9><0><0>
00 4D 45 54 53 59 53 20 6C 61 67 65 4C 20     03ABC C.ATY:  .ASCII  \ Legal SYSTEM\<0><0><0>
                                  00 00       03ACA
6E 69 72 74 73 09 09 09 45 43 49 56 45 44     03ACC C.ATZ:  .ASCII  \DEVICE\<9><9><9>\string\<0>
                                  00 67       03ADA
6F 77 79 65 6B 09 09 09 45 43 52 55 4F 53     03ADC C.AUA:  .ASCII  \SOURCE\<9><9><9>\keyword\
                                  64 72       03AEA
6F 77 79 65 6B 09 09 09 54 45 47 52 41 54     03AEC C.AUB:  .ASCII  \TARGET\<9><9><9>\keyword\
                                  64 72       03AFA
                              00 00 09 09     03AFC C.AUC:  .ASCII  <9><9><0><0>
45 54 53 59 53 20 74 6E 65 72 72 75 43 20     03B00 C.AUD:  .ASCII  \ Current SYSTEM\<0>
                                  00 4D       03B0E
41 20 4D 45 54 53 59 53 20 72 65 74 6E 45     03B10 C.AUE:  .ASCII  \Enter SYSTEM Attribute\<9><9>\(Keyword)\-
79 65 4B 28 09 09 65 74 75 62 69 72 74 74     03B1E                 <0><0><0>
                  00 00 00 29 64 72 6F 77     03B2C
                              00 5D 2D 5B     03B34 C.AUF:  .ASCII  \[-]\<0>
```

```
                                                  00 20 3A 09     03B38 C.AUG:  .ASCII  <9>\: \<0>
65 74 6E 65 20 65 75 6C 61 76 20 65 68 54         03B3C C.AUH:  .ASCII  \The value entered will be put into the D\-
75 70 20 65 62 20 6C 6C 69 77 20 64 65 72         03B4A                 \efinition.\<0><0>
66 65 44 20 65 68 74 20 6F 74 6E 69 20 74         03B58
                  00 00 2E 6E 6F 69 74 69 6E 69   03B66
68 20 79 79 79 79 2D 6D 6D 6D 2D 64 64 28         03B70 C.AUI:  .ASCII  \(dd-mmm-yyyy hh:mm:ss.cc)\<0><0><0>
00 00 00 29 63 63 2E 73 73 3A 6D 6D 3A 68         03B7E
20 72 65 64 6E 69 6C 79 63 5F 79 6E 41 28         03B8C C.AUJ:  .ASCII  \(Any_cylinder Cylinder File_ID File_name\
5F 65 6C 69 46 20 72 65 64 6E 69 6C 79 43         03B9A
      65 6D 61 6E 5F 65 6C 69 46 20 44 49         03BA8
20 65 6E 6F 4E 20 6C 61 63 69 67 6F 4C 20         03BB4 C.AUK:  .ASCII  \ Logical None Virtual)\<0><0>
                  00 00 29 6C 61 75 74 72 69 56   03BC2
4E 4F 49 54 49 53 4F 50 20 72 65 74 6E 45         03BCC C.AUL:  .ASCII  \Enter POSITION qualifier\<9>\(\<0><0>
00 00 28 09 72 65 69 66 69 6C 61 75 71 20         03BDA
6F 66 20 65 75 6C 61 76 20 72 65 74 6E 45         03BE8 C.AUM:  .ASCII  \Enter value for this Secondary\<9>\(\
61 64 6E 6F 63 65 53 20 73 69 68 74 20 72         03BF6
                                  28 09 79 72     03C04
74 61 6C 65 52 20 64 65 78 65 64 6E 49 28         03C08 C.AUN:  .ASCII  \(Indexed Relative Sequential)\<0><0><0>
6C 61 69 74 6E 65 75 71 65 53 20 65 76 69         03C16
                                  00 00 00 29     03C24
53 52 20 45 2F 53 54 53 52 20 53 41 49 28         03C28 C.AUO:  .ASCII  \(IAS RSTS/E RSX-11M RSX-11M-PLUS RT-11 V\-
2D 4D 31 31 2D 58 53 52 20 4D 31 31 2D 58         03C36                 \AX/VMS)\<0>
58 41 56 20 31 31 2D 54 52 20 53 55 4C 50         03C44
                        00 29 53 4D 56 2F         03C52
72 65 76 6F 63 65 72 5F 6E 69 5F 66 49 28         03C58 C.AUP:  .ASCII  \(If_in_recovery_unit Necessary_to_write \-
61 73 73 65 63 65 4E 20 74 69 6E 75 5F 79         03C66                 \Never_RU_journal None)\<0><0>
65 4E 20 65 74 69 72 77 5F 6F 74 5F 79 72         03C74
6C 61 6E 72 75 6F 6A 5F 55 52 5F 72 65 76         03C82
                  00 00 29 65 6E 6F 4E 20         03C90
75 74 65 72 5F 65 67 61 69 72 72 61 43 28         03C98 C.AUQ:  .ASCII  \(Carriage_return FORTRAN None Print)\
6E 6F 4E 20 4E 41 52 54 52 4F 46 20 6E 72         03CA6
                  29 74 6E 69 72 50 20 65         03CB4
20 6D 61 65 72 74 53 20 64 65 78 69 46 28         03CBC C.AUR:  .ASCII  \(Fixed Stream Stream_CR Stream_LF\<0>-
65 72 74 53 20 52 43 5F 6D 61 65 72 74 53         03CCA                 <0><0>
                  00 00 00 46 4C 5F 6D 61         03CD8
72 61 56 20 64 65 6E 69 66 65 64 6E 55 20         03CE0 C.AUS:  .ASCII  \ Undefined Variable VFC)\
            29 43 46 56 20 65 6C 62 61 69         03CEE
6E 69 42 20 34 6E 69 42 20 32 6E 69 42 28         03CF8 C.AUT:  .ASCII  \(Bin2 Bin4 Bin8 Decimal Int2 Int4 Int8 S\-
32 74 6E 49 20 6C 61 6D 69 63 65 44 20 38         03D06                 \tring)\<0><0>
72 74 53 20 38 74 6E 49 20 34 74 6E 49 20         03D14
                        00 00 29 67 6E 69         03D22
6F 66 20 65 75 6C 61 76 20 72 65 74 6E 45         03D28 C.AUU:  .ASCII  \Enter value for this Secondary\<9>\(Keyw\-
61 64 6E 6F 63 65 53 20 73 69 68 74 20 72         03D36                 \ord)\
      29 64 72 6F 77 79 65 4B 28 09 79 72         03D44
                                  00 5D 2D 5B     03D50 C.AUV:  .ASCII  \[-]\<0>
                                  00 20 3A 09     03D54 C.AUW:  .ASCII  <9>\: \<0>
                  4C000000  00000000  00000000  00000000      03D58 C.AUX:  .LONG   0,0,0,^X4C000000
                                              00 00 00        03D68         .BYTE   0,0,0
                                                              03D6B         .BLKB   1
                        29 30 30 31 3C 73 62 41               03D6C C.AUY:  .ASCII  \Abs<100)\
                        29 61 67 69 47 31 2D 30               03D74 C.AUZ:  .ASCII  \0-1Giga)\
                                    00 00 2D 30               03D7C C.AVA:  .ASCII  \0-\<0><0>
                                    00 5D 2D 5B               03D80 C.AVB:  .ASCII  \[-]\<0>
                                    00 20 3A 20               03D84 C.AVC:  .ASCII  \ : \<0>
                                    00 20 3A 09               03D88 C.AVD:  .ASCII  <9>\: \<0>
                  4C000000  00000000  00000000  00000000      03D8C C.AVE:  .LONG   0,0,0,^X4C000000
                                              00 00 00        03D9C         .BYTE   0,0,0
```

```
                                                                03D9F                   .BLKB   1
                00  00  00  29  72  74  73  2D  65  74  61  44  03DA0 C.AVF:  .ASCII  \Date-str)\<0><0><0>
                                                00  5D  2D  5B  03DAC C.AVG:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03DB0 C.AVH:  .ASCII  <9>\: \<0>
 75  6E  5B  29  73  72  61  68  63  20  32  33  2D  31          03DB4 C.AVI:  .ASCII  \1-32 chars)[null]\<9>\: \
                                            20  3A  09  5D  6C  6C  03DC2
 6E  5B  29  73  72  61  68  63  20  36  32  31  2D  31          03DC8 C.AVJ:  .ASCII  \1-126 chars)[null]\
                                            5D  6C  6C  75  03DD6
                                                20  3A  03DDA C.AVK:  .ASCII  \: \
                        00  29  6F  4E  2F  7?  65  59  03DDC C.AVL:  .ASCII  \Yes/No)\<0>
                                                00  5D  2D  5B  03DE4 C.AVM:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03DE8 C.AVN:  .ASCII  <9>\: \<0>
                        29  72  74  73  2D  43  49  55  03DEC C.AVO:  .ASCII  \UIC-str)\
                                                00  5D  2D  5B  03DF4 C.AVP:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03DF8 C.AVQ:  .ASCII  <9>\: \<0>
                00  00  00  29  72  74  73  2D  74  6F  72  50  03DFC C.AVR:  .ASCII  \Prot-str)\<0><0><0>
                                                00  5D  2D  5B  03E08 C.AVS:  .ASCII  \[-]\<0>
                                                00  00  20  3A  03E0C C.AVT:  .ASCII  \: \<0><0>
                        29  64  72  6F  77  79  65  4B  03E10 C.AVU:  .ASCII  \Keyword)\
                                                00  5D  2D  5B  03E18 C.AVV:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03E1C C.AVW:  .ASCII  <9>\: \<0>
        00000000  00000000  00000000  00000000  00000041  03E20 C.AVX:  .LONG   ^X41,0,0,0,0,0,0,0
                  00000000  00000000  00000000  00000000  03E34
 4E  4F  49  54  49  53  4F  50  20  72  65  74  6E  45      03E40 C.AVY:  .ASCII  \Enter POSITION value\<9><9>\(\<0>
                00  28  09  09  65  75  6C  61  76  20      03E4E
                        29  61  67  69  47  31  2D  30  03E58 C.AVZ:  .ASCII  \0-1Giga)\
                                                00  5D  2D  5B  03E60 C.AWA:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03E64 C.AWB:  .ASCII  <9>\: \<0>
                        29  72  74  73  2D  44  49  46  03E68 C.AWC:  .ASCII  \FID-str)\
                                                00  5D  2D  5B  03E70 C.AWD:  .ASCII  \[-]\<0>
                                                00  20  3A  09  03E74 C.AWE:  .ASCII  <9>\: \<0>
 6E  5B  29  73  72  61  68  63  20  39  30  31  2D  31          03E78 C.AWF:  .ASCII  \1-109 chars)[null]\
                                            5D  6C  6C  75  03E86
                                                20  3A  03E8A C.AWG:  .ASCII  \: \
                00  29  6D  75  6E  2F  27  72  61  68  63  27  03E8C C.AWH:  .ASCII  \'char'/num)\<0>
                                                00  5D  2D  5B  03E98 C.AWI:  .ASCII  \[-]\<0>
                                                    20  3A  09  03E9C C.AWJ:  .ASCII  <9>\: \

                                                            00000 WRITE_HELP:                                       ; 0190
                                                        0000 00000          .WORD   ^M<>
                        32          13 00000000G    EF  CF 00002          CASEL   QTAB_OFFSET,#19,#50             ; 0194
                                                0000V       0000A          .DISPL  17$
                                                0000V       0000C          .DISPL  17$
                                                0000V       0000E          .DISPL  17$
                                                0000V       00010          .DISPL  19$
                                                0000V       00012          .DISPL  11$
                                                0000V       00014          .DISPL  14$
                                                0066        00016          .DISPL  102
                                                0000V       00018          .DISPL  6$
                                                0000V       0001A          .DISPL  7$
                                                0000V       0001C          .DISPL  4$
                                                0000V       0001E          .DISPL  3$
                                                0000V       00020          .DISPL  5$
                                                0066        00022          .DISPL  102
                                                0000V       00024          .DISPL  18$
                                                0000V       00026          .DISPL  32$
                                                0000V       00028          .DISPL  13$
```

```
                    0000V     0002A          .DISPL  13$
                    0000V     0002C          .DISPL  13$
                    0000V     0002E          .DISPL  20$
                    0000V     00030          .DISPL  21$
                    0000V     00032          .DISPL  39$
                    0000V     00034          .DISPL  34$
                    0066      00036          .DISPL  102
                    0000V     00038          .DISPL  1$
                    0000V     0003A          .DISPL  26$
                    0000V     0003C          .DISPL  26$
                    0000V     0003E          .DISPL  26$
                    0066      00040          .DISPL  102
                    0066      00042          .DISPL  102
                    0000V     00044          .DISPL  12$
                    0000V     00046          .DISPL  12$
                    0000V     00048          .DISPL  12$
                    0000V     0004A          .DISPL  2$
                    0000V     0004C          .DISPL  15$
                    0000V     0004E          .DISPL  15$
                    0000V     00050          .DISPL  15$
                    0000V     00052          .DISPL  27$
                    0000V     00054          .DISPL  22$
                    0000V     00056          .DISPL  36$
                    0000V     00058          .DISPL  35$
                    0066      0005A          .DISPL  102
                    0000V     0005C          .DISPL  33$
                    0000V     0005E          .DISPL  10$
                    0000V     00060          .DISPL  16$
                    0000V     00062          .DISPL  9$
                    0000V     00064          .DISPL  31$
                    0000V     00066          .DISPL  8$
                    0066      00068          .DISPL  102
                    0066      0006A          .DISPL  102
                    0000V     0006C          .DISPL  35$
                    0000V     0006E          .DISPL  35$
                    0000V  31 00070          BRW     40$
          00000000G   EF  9F 00073 1$:       PUSHAB  SHIFT                          ; 0198
                      04  DD 00079          PUSHL   #4
          00000000G   EF  9F 0007B          PUSHAB  PAS$FV_OUTPUT
00000000G EF              03  FB 00081          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0D3  EF  9F 00088          PUSHAB  C.AAA
                      32  DD 0008E          PUSHL   #50
          00000000G   EF  9F 00090          PUSHAB  PAS$FV_OUTPUT
00000000G EF              03  FB 00096          CALLS   #3,PAS$WRITE_STRING
          00000000G   EF  9F 0009D          PUSHAB  PAS$FV_OUTPUT
00000000G EF              01  FB 000A3          CALLS   #1,PAS$WRITELN2
                    G000V  31 000AA          BRW     41$
          00000000G   EF  9F 000AD 2$:       PUSHAB  SHIFT                          ; 0202
                      04  DD 000B3          PUSHL   #4
          00000000G   EF  9F 000B5          PUSHAB  PAS$FV_OUTPUT
00000000G EF              03  FB 000BB          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0CD  EF  9F 000C2          PUSHAB  C.AAB
                      34  DD 000C8          PUSHL   #52
          00000000G   EF  9F 000CA          PUSHAB  PAS$FV_OUTPUT
00000000G EF              03  FB 000D0          CALLS   #3,PAS$WRITE_STRING
          00000000G   EF  9F 000D7          PUSHAB  PAS$FV_OUTPUT
00000000G EF              01  FB 000DD          CALLS   #1,PAS$WRITELN2
```

```
                          0000V  31 000E4          BRW     41$
                 00000000G EF    9F 000E7 3$:     PUSHAB  SHIFT                          ; 0207
                           04    DD 000ED         PUSHL   #4
                 00000000G EF    9F 000EF         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 000F5     CALLS   #3,PAS$WRITE_STRING
                 FFFFC0C7  EF    9F 000FC         PUSHAB  C.AAC
                           2C    DD 00102         PUSHL   #44
                 00000000G EF    9F 00104         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 0010A     CALLS   #3,PAS$WRITE_STRING
                 00000000G EF    9F 00111         PUSHAB  CRLF_SHIFT
                           06    DD 00117         PUSHL   #6
                 00000000G EF    9F 00119         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 0011F     CALLS   #3,PAS$WRITE_STRING
                 FFFFC0C9  EF    9F 00126         PUSHAB  C.AAD
                           22    DD 0012C         PUSHL   #34
                 00000000G EF    9F 0012E         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 00134     CALLS   #3,PAS$WRITE_STRING
                 00000000G EF    9F 0013B         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               01 FB 00141     CALLS   #1,PAS$WRITELN2
                          0000V  31 00148          BRW     41$
                 00000000G EF    9F 0014B 4$:     PUSHAB  SHIFT                          ; 0213
                           04    DD 00151         PUSHL   #4
                 00000000G EF    9F 00153         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 00159     CALLS   #3,PAS$WRITE_STRING
                 FFFFC0B3  EF    9F 00160         PUSHAB  C.AAE
                           28    DD 00166         PUSHL   #40
                 00000000G EF    9F 00168         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 0016E     CALLS   #3,PAS$WRITE_STRING
                 00000000G EF    9F 00175         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               01 FB 0017B     CALLS   #1,PAS$WRITELN2
                          0000V  31 00182          BRW     41$
                 00000000G EF    9F 00185 5$:     PUSHAB  SHIFT                          ; 0217
                           04    DD 0018B         PUSHL   #4
                 00000000G EF    9F 0018D         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 00193     CALLS   #3,PAS$WRITE_STRING
                 FFFFC0A1  EF    9F 0019A         PUSHAB  C.AAF
                           28    DD 001A0         PUSHL   #40
                 00000000G EF    9F 001A2         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 001A8     CALLS   #3,PAS$WRITE_STRING
                 00000000G EF    9F 001AF         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               01 FB 001B5     CALLS   #1,PAS$WRITELN2
                          0000V  31 001BC          BRW     41$
                 00000000G EF    9F 001BF 6$:     PUSHAB  SHIFT                          ; 0221
                           04    DD 001C5         PUSHL   #4
                 00000000G EF    9F 001C7         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 001CD     CALLS   #3,PAS$WRITE_STRING
                 FFFFC08F  EF    9F 001D4         PUSHAB  C.AAG
                           2D    DD 001DA         PUSHL   #45
                 00000000G EF    9F 001DC         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 001E2     CALLS   #3,PAS$WRITE_STRING
                 00000000G EF    9F 001E9         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               01 FB 001EF     CALLS   #1,PAS$WRITELN2
                          0000V  31 001F6          BRW     41$
                 00000000G EF    9F 001F9 7$:     PUSHAB  SHIFT                          ; 0225
                           04    DD 001FF         PUSHL   #4
                 00000000G EF    9F 00201         PUSHAB  PAS$FV_OUTPUT
00000000G        EF               03 FB 00207     CALLS   #3,PAS$WRITE_STRING
```

```
                                Generated Code

                                FFFFC085   EF   9F 0020E          PUSHAB  C.AAH
                                           31   DD 00214          PUSHL   #49
                                00000000G  EF   9F 00216          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0021C          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00223          PUSHAB  CRLF_SHIFT
                                           06   DD 00229          PUSHL   #6
                                00000000G  EF   9F 0022B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00231          CALLS   #3,PAS$WRITE_STRING
                                FFFFC08F   EF   9F 00238          PUSHAB  C.AAI
                                           2D   DD 0023E          PUSHL   #45
                                00000000G  EF   9F 00240          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00246          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 0024D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              01   FB 00253          CALLS   #1,PAS$WRITELN2
                                      0000V 31   0025A          BRW     41$
                                00000000G  EF   9F 0025D  8$:     PUSHAB  SHIFT                                   ; 0231
                                           04   DD 00263          PUSHL   #4
                                00000000G  EF   9F 00265          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0026B          CALLS   #3,PAS$WRITE_STRING
                                FFFFC085   EF   9F 00272          PUSHAB  C.AAJ
                                           38   DD 00278          PUSHL   #56
                                00000000G  EF   9F 0027A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00280          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00287          PUSHAB  CRLF_SHIFT
                                           06   DD 0028D          PUSHL   #6
                                00000000G  EF   9F 0028F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00295          CALLS   #3,PAS$WRITE_STRING
                                FFFFC093   EF   9F 0029C          PUSHAB  C.AAK
                                           19   DD 002A2          PUSHL   #25
                                00000000G  EF   9F 002A4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 002AA          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 002B1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              01   FB 002B7          CALLS   #1,PAS$WRITELN2
                                      0000V 31   002BE          BRW     41$
                                00000000G  EF   9F 002C1  9$:     PUSHAB  SHIFT                                   ; 0237
                                           04   DD 002C7          PUSHL   #4
                                00000000G  EF   9F 002C9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 002CF          CALLS   #3,PAS$WRITE_STRING
                                FFFFC075   EF   9F 002D6          PUSHAB  C.AAL
                                           2F   DD 002DC          PUSHL   #47
                                00000000G  EF   9F 002DE          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 002E4          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 002EB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              01   FB 002F1          CALLS   #1,PAS$WRITELN2
                                      0000V 31   002F8          BRW     41$
                                00000000G  EF   9F 002FB  10$:    PUSHAB  SHIFT                                   ; 0241
                                           04   DD 00301          PUSHL   #4
                                00000000G  EF   9F 00303          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00309          CALLS   #3,PAS$WRITE_STRING
                                FFFFC06B   EF   9F 00310          PUSHAB  C.AAM
                                           2F   DD 00316          PUSHL   #47
                                00000000G  EF   9F 00318          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0031E          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00325          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              01   FB 0032B          CALLS   #1,PAS$WRITELN2
                                      0000V 31   00332          BRW     41$
                                00000000G  EF   9F 00335  11$:    PUSHAB  SHIFT                                   ; 0245
```

```
                                 04   DD 0033B          PUSHL   #4
                    00000000G    EF   9F 0033D          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 00343          CALLS   #3,PAS$WRITE_STRING
                    FFFFC061     EF   9F 0034A          PUSHAB  C.AAN
                                 37   DD 00350          PUSHL   #55
                    00000000G    EF   9F 00352          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 00358          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 0035F          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  01   FB 00365          CALLS   #1,PAS$WRITELN2
                              0000V   31 0036C          BRW     41$
                    00000000G    EF   9F 0036F  12$:     PUSHAB  SHIFT                                   ; 0252
                                 04   DD 00375          PUSHL   #4
                    00000000G    EF   9F 00377          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 0037D          CALLS   #3,PAS$WRITE_STRING
                    FFFFC05F     EF   9F 00384          PUSHAB  C.AAO
                                 35   DD 0038A          PUSHL   #53
                    00000000G    EF   9F 0038C          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 00392          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00399          PUSHAB  CRLF_SHIFT
                                 06   DD 0039F          PUSHL   #6
                    00000000G    EF   9F 003A1          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 003A7          CALLS   #3,PAS$WRITE_STRING
                    FFFFC06D     EF   9F 003AE          PUSHAB  C.AAP
                                 37   DD 003B4          PUSHL   #55
                    00000000G    EF   9F 003B6          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 003BC          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 003C3          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  01   FB 003C9          CALLS   #1,PAS$WRITELN2
                              0000V   31 003D0          BRW     41$
                    00000000G    EF   9F 003D3  13$:     PUSHAB  SHIFT                                   ; 0260
                                 04   DD 003D9          PUSHL   #4
                    00000000G    EF   9F 003DB          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 003E1          CALLS   #3,PAS$WRITE_STRING
                    FFFFC06B     EF   9F 003E8          PUSHAB  C.AAQ
                                 38   DD 003EE          PUSHL   #56
                    00000000G    EF   9F 003F0          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 003F6          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 003FD          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  01   FB 00403          CALLS   #1,PAS$WRITELN2
                              0000V   31 0040A          BRW     41$
                    00000000G    EF   9F 0040D  14$:     PUSHAB  SHIFT                                   ; 0265
                                 04   DD 00413          PUSHL   #4
                    00000000G    EF   9F 00415          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 0041B          CALLS   #3,PAS$WRITE_STRING
                    FFFFC069     EF   9F 00422          PUSHAB  C.AAR
                                 3A   DD 00428          PUSHL   #58
                    00000000G    EF   9F 0042A          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 00430          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00437          PUSHAB  CRLF_SHIFT
                                 06   DD 0043D          PUSHL   #6
                    00000000G    EF   9F 0043F          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 00445          CALLS   #3,PAS$WRITE_STRING
                    FFFFC07B     EF   9F 0044C          PUSHAB  C.AAS
                                 34   DD 00452          PUSHL   #52
                    00000000G    EF   9F 00454          PUSHAB  PAS$FV_OUTPUT
00000000G    EF                  03   FB 0045A          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00461          PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF              01   FB 00467          CALLS   #1,PAS$WRITELN2
                     0000V 31   00046E           BRW     41$
                00000000G  EF   9F 00471 15$:     PUSHAB  SHIFT                                   ; 0273
                           04   DD 00477          PUSHL   #4
                00000000G  EF   9F 00479          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0047F          CALLS   #3,PAS$WRITE_STRING
                FFFFC075   EF   9F 00486          PUSHAB  C.AAT
                           31   DD 0048C          PUSHL   #49
                00000000G  EF   9F 0048E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00494          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0049B          PUSHAB  CRLF_SHIFT
                           06   DD 004A1          PUSHL   #6
                00000000G  EF   9F 004A3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004A9          CALLS   #3,PAS$WRITE_STRING
                FFFFC07F   EF   9F 004B0          PUSHAB  C.AAU
                           3D   DD 004B6          PUSHL   #61
                00000000G  EF   9F 004B8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004BE          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 004C5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              01   FB 004CB          CALLS   #1,PAS$WRITELN2
                     0000V 31   004D2            BRW     41$
                00000000G  EF   9F 004D5 16$:     PUSHAB  SHIFT                                   ; 0279
                           04   DD 004DB          PUSHL   #4
                00000000G  EF   9F 004DD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004E3          CALLS   #3,PAS$WRITE_STRING
                FFFFC085   EF   9F 004EA          PUSHAB  C.AAV
                           34   DD 004F0          PUSHL   #52
                00000000G  EF   9F 004F2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004F8          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 004FF          PUSHAB  CRLF_SHIFT
                           06   DD 00505          PUSHL   #6
                00000000G  EF   9F 00507          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0050D          CALLS   #3,PAS$WRITE_STRING
                FFFFC08F   EF   9F 00514          PUSHAB  C.AAW
                           33   DD 0051A          PUSHL   #51
                00000000G  EF   9F 0051C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00522          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 00529          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              01   FB 0052F          CALLS   #1,PAS$WRITELN2
                     0000V 31   00536            BRW     41$
                00000000G  EF   9F 00539 17$:     PUSHAB  SHIFT                                   ; 0288
                           04   DD 0053F          PUSHL   #4
                00000000G  EF   9F 00541          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00547          CALLS   #3,PAS$WRITE_STRING
                FFFFC089   EF   9F 0054E          PUSHAB  C.AAX
                           3A   DD 00554          PUSHL   #58
                00000000G  EF   9F 00556          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0055C          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 00563          PUSHAB  CRLF_SHIFT
                           06   DD 00569          PUSHL   #6
                00000000G  EF   9F 0056B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00571          CALLS   #3,PAS$WRITE_STRING
                FFFFC09B   EF   9F 00578          PUSHAB  C.AAY
                           3D   DD 0057E          PUSHL   #61
                00000000G  EF   9F 00580          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00586          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0058D          PUSHAB  PAS$FV_OUTPUT
```

```
                                 01   FB 00593          CALLS   #1,PAS$WRITELN2
                           0000V 31   0059A          BRW     41$
                    00000000G  EF   9F 0059D 18$:     PUSHAB  SHIFT                                   ; 0295
                                 04   DD 005A3          PUSHL   #4
                    00000000G  EF   9F 005A5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 005AB          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0A1   EF   9F 005B2          PUSHAB  C.AAZ
                                 35   DD 005B8          PUSHL   #53
                    00000000G  EF   9F 005BA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 005C0          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 005C7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    01   FB 005CD          CALLS   #1,PAS$WRITELN2
                           0000V 31   005D4          BRW     41$
                    00000000G  EF   9F 005D7 19$:     PUSHAB  SHIFT                                   ; 0300
                                 04   DD 005DD          PUSHL   #4
                    00000000G  EF   9F 005DF          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 005E5          CALLS   #3,PAS$WRITE_STRING
                    FFFFC09F   EF   9F 005EC          PUSHAB  C.ABA
                                 33   DD 005F2          PUSHL   #51
                    00000000G  EF   9F 005F4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 005FA          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00601          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    01   FB 00607          CALLS   #1,PAS$WRITELN2
                           0000V 31   0060E          BRW     41$
                    00000000G  EF   9F 00611 20$:     PUSHAB  SHIFT                                   ; 0305
                                 04   DD 00617          PUSHL   #4
                    00000000G  EF   9F 00619          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0061F          CALLS   #3,PAS$WRITE_STRING
                    FFFFC099   EF   9F 00626          PUSHAB  C.ABB
                                 3A   DD 0062C          PUSHL   #58
                    00000000G  EF   9F 0062E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00634          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0063B          PUSHAB  CRLF_SHIFT
                                 06   DD 00641          PUSHL   #6
                    00000000G  EF   9F 00643          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00649          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0AB   EF   9F 00650          PUSHAB  C.ABC
                                 38   DD 00656          PUSHL   #56
                    00000000G  EF   9F 00658          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0065E          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00665          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    01   FB 0066B          CALLS   #1,PAS$WRITELN2
                           0000V 31   00672          BRW     41$
                    00000000G  EF   9F 00675 21$:     PUSHAB  SHIFT                                   ; 0312
                                 04   DD 0067B          PUSHL   #4
                    00000000G  EF   9F 0067D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00683          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0A9   EF   9F 0068A          PUSHAB  C.ABD
                                 34   DD 00690          PUSHL   #52
                    00000000G  EF   9F 00692          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00698          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0069F          PUSHAB  CRLF_SHIFT
                                 06   DD 006A5          PUSHL   #6
                    00000000G  EF   9F 006A7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 006AD          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0B3   EF   9F 006B4          PUSHAB  C.ABE
                                 32   DD 006BA          PUSHL   #50
```

```
                    00000000G  EF   9F 006BC          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 006C2          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 006C9          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           01 FB 006CF          CALLS   #1,PAS$WRITELN2
                           0000V 31 006D6          BRW     41$
     00V00000000G   EF           00 E1 006D9 22$:     BBC     #0,WAIT_HELP,24$                        ; 0319
                    00000000G  EF   9F 006E1          PUSHAB  SHIFT                                   ; 0321
                               04 DD 006E7          PUSHL   #4
                    00000000G  EF   9F 006E9          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 006EF          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0A5   EF   9F 006F6          PUSHAB  C.ABF
                               34 DD 006FC          PUSHL   #52
                    00000000G  EF   9F 006FE          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 00704          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0070B          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           01 FB 00711          CALLS   #1,PAS$WRITELN2
                           0000V 31 00718          BRW     41$
                    00000000G  EF   9F 0071B 24$:     PUSHAB  SHIFT                                   ; 0326
                               04 DD 00721          PUSHL   #4
                    00000000G  EF   9F 00723          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 00729          CALLS   #3,PAS$WRITE_STRING
                    FFFFC09F   EF   9F 00730          PUSHAB  C.ABG
                               39 DD 00736          PUSHL   #57
                    00000000G  EF   9F 00738          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 0073E          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00745          PUSHAB  CRLF_SHIFT
                               06 DD 0074B          PUSHL   #6
                    00000000G  EF   9F 0074D          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 00753          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0B1   EF   9F 0075A          PUSHAB  C.ABH
                               3B DD 00760          PUSHL   #59
                    00000000G  EF   9F 00762          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 00768          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0076F          PUSHAB  CRLF_SHIFT
                               06 DD 00775          PUSHL   #6
                    00000000G  EF   9F 00777          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 0077D          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0C3   EF   9F 00784          PUSHAB  C.ABI
                               3C DD 0078A          PUSHL   #60
                    00000000G  EF   9F 0078C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 00792          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00799          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           01 FB 0079F          CALLS   #1,PAS$WRITELN2
                           0000V 31 007A6          BRW     41$
                    00000000G  EF   9F 007A9 26$:     PUSHAB  SHIFT                                   ; 0337
                               04 DD 0C7AF          PUSHL   #4
                    00000000G  EF   9F 007B1          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 007B7          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0C5   EF   9F 007BE          PUSHAB  C.ABJ
                               2D DD 007C4          PUSHL   #45
                    00000000G  EF   9F 007C6          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03 FB 007CC          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 007D3          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           01 FB 007D9          CALLS   #1,PAS$WRITELN2
                           0000V 31 007E0          BRW     41$
     00V00000000G   EF           00 E1 007E3 27$:     BBC     #0,WAIT_HELP,29$                        ; 0343
                    00000000G  EF   9F 007EB          PUSHAB  SHIFT                                   ; 0345
```

```
                                 04    DD 007F1          PUSHL   #4
                    00000000G    EF    9F 007F3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 007F9          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0B3     EF    9F 00800          PUSHAB  C.ABK
                                 39    DD 00806          PUSHL   #57
                    00000000G    EF    9F 00808          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0080E          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 00815          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01    FB 0[illegible]1B  CALLS   #1,PAS$WRITELN2
                              0000V  31    00822          BRW     41$
                    00000000G    EF    9F 00825 29$:     PUSHAB  SHIFT                                   ; 0350
                                 04    DD 0082B          PUSHL   #4
                    00000000G    EF    9F 0082D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00833          CALLS   #3,PAS$WRITF_STRING
                    FFFFC0B5     EF    9F 0083A          PUSHAB  C.ABL
                                 03    DD 00840          PUSHL   #3
                    00000000G    EF    9F 00842          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00848          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 0084F          PUSHAB  CRLF_SHIFT
                                 06    DD 00855          PUSHL   #6
                    00000000G    EF    9F 00857          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0085D          CALLS   #3,PAS$WRITE_STRING
                    FFFFC08F     EF    9F 00864          PUSHAB  C.ABM
                                 39    DD 0086A          PUSHL   #57
                    00000000G    EF    9F 0086C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00872          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 00879          PUSHAB  CRLF_SHIFT
                                 06    DD 0087F          PUSHL   #6
                    00000000G    EF    9F 00881          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00887          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0A1     EF    9F 0088E          PUSHAB  C.ABN
                                 36    DD 00894          PUSHL   #54
                    00000000G    EF    9F 00896          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0089C          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 008A3          PUSHAB  CRLF_SHIFT
                                 06    DD 008A9          PUSHL   #6
                    00000000G    EF    9F 008AB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 008B1          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0AF     EF    9F 008B8          PUSHAB  C.ABO
                                 37    DD 008BE          PUSHL   #55
                    00000000G    EF    9F 008C0          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 008C6          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 008CD          PUSHAB  CRLF_SHIFT
                                 06    DD 008D3          PUSHL   #6
                    00000000G    EF    9F 008D5          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 008DB          CALLS   #3,PAS$WRITE_STRING
                    FFFFC0BD     EF    9F 008E2          PUSHAB  C.ABP
                                 39    DD 008E8          PUSHL   #57
                    00000000G    EF    9F 008EA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 008F0          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 008F7          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01    FB 008FD          CALLS   #1,PAS$WRITELN2
                              0000V  31    00904          BRW     41$
                    00000000G    EF    9F 00907 31$:     PUSHAB  SHIFT                                   ; 0365
                                 04    DD 0090D          PUSHL   #4
                    00000000G    EF    9F 0090F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00915          CALLS   #3,PAS$WRITE_STRING
```

```
                 FFFFC0BF    EF    9F 0091C          PUSHAB  C.ABQ
                             29    DD 00922          PUSHL   #41
                 00000000G   EF    9F 00924          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 0092A          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 00931          PUSHAB  CRLF_SHIFT
                             06    DD 00937          PUSHL   #6
                 00000000G   EF    9F 00939          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 0093F          CALLS   #3,PAS$WRITE_STRING
                 FFFFC0C1    EF    9F 00946          PUSHAB  C.ABR
                             3C    DD 0094C          PUSHL   #60
                 00000000G   EF    9F 0094E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 00954          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 0095B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               01    FB 00961          CALLS   #1,PAS$WRITELN2
                          0000V    31 00968          BRW     41$
                 00000000G   EF    9F 0096B 32$:     PUSHAB  SHIFT                                   ; 0371
                             04    DD 00971          PUSHL   #4
                 00000000G   EF    9F 00973          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 00979          CALLS   #3,PAS$WRITE_STRING
                 FFFFC0C3    EF    9F 00980          PUSHAB  C.ABS
                             1E    DD 00986          PUSHL   #30
                 00000000G   EF    9F 00988          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 0098E          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 00995          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               01    FB 0099B          CALLS   #1,PAS$WRITELN2
                          0000V    31 009A2          BRW     41$
                 00000000G   EF    9F 009A5 33$:     PUSHAB  SHIFT                                   ; 0375
                             04    DD 009AB          PUSHL   #4
                 00000000G   EF    9F 009AD          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 009B3          CALLS   #3,PAS$WRITE_STRING
                 FFFFC0A9    EF    9F 009BA          PUSHAB  C.ABT
                             2C    DD 009C0          PUSHL   #44
                 00000000G   EF    9F 009C2          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 009C8          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 009CF          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               01    FB 009D5          CALLS   #1,PAS$WRITELN2
                          0000V    31 009DC          BRW     41$
                 00000000G   EF    9F 009DF 34$:     PUSHAB  SHIFT                                   ; 0379
                             04    DD 009E5          PUSHL   #4
                 00000000G   EF    9F 009E7          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 009ED          CALLS   #3,PAS$WRITE_STRING
                 FFFFC09B    EF    9F 009F4          PUSHAB  C.ABU
                             2F    DD 009FA          PUSHL   #47
                 00000000G   EF    9F 009FC          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 00A02          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 00A09          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               01    FB 00A0F          CALLS   #1,PAS$WRITELN2
                          0000V    31 00A16          BRW     41$
                 00000000G   EF    9F 00A19 35$:     PUSHAB  SHIFT                                   ; 0385
                             04    DD 00A1F          PUSHL   #4
                 00000000G   EF    9F 00A21          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 00A27          CALLS   #3,PAS$WRITE_STRING
                 FFFFC091    EF    9F 00A2E          PUSHAB  C.ABV
                             2C    DD 00A34          PUSHL   #44
                 00000000G   EF    9F 00A36          PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03    FB 00A3C          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF    9F 00A43          PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF              01  FB  00A49          CALLS   #1,PAS$WRITELN2
                     0000V 31  00A50          BRW     41$
                00000000G  EF  9F  00A53  36$:    PUSHAB  SHIFT                                   ; 0391
                           04  DD  00A59          PUSHL   #4
                00000000G  EF  9F  00A5B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03  FB  00A61          CALLS   #3,PAS$WRITE_STRING
                FFFFC083   EF  9F  00A68          PUSHAB  C.ABW
                           3C  DD  00A6E          PUSHL   #60
                00000000G  EF  9F  00A70          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03  FB  00A76          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F  00A7D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              01  FB  00A83          CALLS   #1,PAS$WRITELN2
           03  00000108G  EF  D1  00A8A          CMPL    IDATA+264,#3                            ; 0394
                     00V   13  00A91          BEQL    38$
                00000000G  EF  9F  00A93          PUSHAB  SHIFT                                   ; 0396
                           04  DD  00A99          PUSHL   #4
                00000000G  EF  9F  00A9B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03  FB  00AA1          CALLS   #3,PAS$WRITE_STRING
                FFFFC07F   EF  9F  00AA8          PUSHAB  C.ABX
                           2C  DD  00AAE          PUSHL   #44
                00000000G  EF  9F  00AB0          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03  FB  00AB6          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F  00ABD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              01  FB  00AC3          CALLS   #1,PAS$WRITELN2
                     00V   11  00ACA  38$:    BRB     41$
                00000000G  EF  9F  00ACC  39$:    PUSHAB  SHIFT                                   ; 0403
                           04  DD  00AD2          PUSHL   #4
                00000000G  EF  9F  00AD4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03  FB  00ADA          CALLS   #3,PAS$WRITE_STRING
                FFFFC072   EF  9F  00AE1          PUSHAB  C.ABY
                           2C  DD  00AE7          PUSHL   #44
                00000000G  EF  9F  00AE9          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF              03  FB  00AEF          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F  00AF6          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF              01  FB  00AFC          CALLS   #1,PAS$WRITELN2
                     00V   11  00B03          BRB     41$
                           00B05  40$:
00V00000000G  EF              00  E1  00B05  41$:    BBC     #0,WAIT_HELP,44$                    ; 0411
00V00000000G  EF              00  E0  00B0D          BBS     #0,AUTO_TUNE,44$
                00004140   8F  DF  00B15          PUSHAF  #^F3.0                                  ; 0417
   00000000G  EF              01  FB  00B1B          CALLS   #1,LIB$WAIT
                              04  00B22  44$:    RET                                             ; 0419

; Routine Size: 2851 bytes,    Routine Base: $CODE + 03E9F

                                    00000  WRITE_QUESTION:                                       ; 0464
                               0000 00000          .WORD   ^M<>
                5E                  04  C2  00002          SUBL2   #4,SP
      3C           0B  00000000G    EF  CF  00005          CASEL   QTAB_OFFSET,#11,#60          ; 0468
                           0000V         0000D          .DISPL  93$
                           0000V         0000F          .DISPL  96$
                           0000V         00011          .DISPL  97$
                           0000V         00013          .DISPL  94$
                           0000V         00015          .DISPL  95$
                           C07A          00017          .DISPL  122
                           007A          00019          .DISPL  122
                           007A          0001B          .DISPL  122
```

```
                0000V     0001D          .DISPL  73$
                0000V     0001F          .DISPL  74$
                0000V     00021          .DISPL  75$
                0000V     00023          .DISPL  78$
                0000V     00025          .DISPL  65$
                0000V     00027          .DISPL  77$
                0000V     00029          .DISPL  92$
                0000V     0002B          .DISPL  48$
                0000V     0002D          .DISPL  49$
                0000V     0002F          .DISPL  43$
                0000V     00031          .DISPL  37$
                0000V     00033          .DISPL  44$
                0000V     00035          .DISPL  25$
                0000V     00037          .DISPL  76$
                0000V     00039          .DISPL  99$
                0000V     0003B          .DISPL  69$
                0000V     0003D          .DISPL  6$
                0000V     0003F          .DISPL  7$
                0000V     00041          .DISPL  83$
                0000V     00043          .DISPL  87$
                0000V     00045          .DISPL  101$
                0000V     00047          .DISPL  108$
                0000V     00049          .DISPL  27$
                0000V     0004B          .DISPL  26$
                0000V     0004D          .DISPL  91$
                0000V     0004F          .DISPL  18$
                0000V     00051          .DISPL  19$
                007A      00053          .DISPL  122
                0000V     00055          .DISPL  134$
                0000V     00057          .DISPL  51$
                0000V     00059          .DISPL  1$
                0000V     0005B          .DISPL  2$
                0000V     0005D          .DISPL  34$
                0000V     0005F          .DISPL  11$
                0000V     00061          .DISPL  12$
                0000V     00063          .DISPL  80$
                0000V     00065          .DISPL  98$
                0000V     00067          .DISPL  61$
                0000V     00069          .DISPL  147$
                0000V     0006B          .DISPL  109$
                007A      0006D          .DISPL  122
                0000V     0006F          .DISPL  100$
                0000V     00071          .DISPL  50$
                0000V     00073          .DISPL  79$
                0000V     00075          .DISPL  33$
                0000V     00077          .DISPL  102$
                0000V     00079          .DISPL  32$
                0000V     0007B          .DISPL  20$
                0000V     0007D           DISPL  142$
                0000V     0007F          .DISPL  13$
                0000V     00081          .DISPL  14$
                0000V     00083          .DISPL  116$
                0000V     00085          .DISPL  154$
                0000V  31 00087          BRW     168$
00000000G       EF  9F 0008A 1$:         PUSHAB  SHIFT                          ; 0472
                    04 DD 00090          PUSHL   #4
00000000G       EF  9F 00092             PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF                03   FB 00098          CALLS   #3,PAS$WRITE_STRING
                FFFFBFBD     EF   9F 0009F          PUSHAB  C.ABZ
                             2F   DD 000A5          PUSHL   #47
                00000000G    EF   9F 000A7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 000AD          CALLS   #3,PAS$WRITE_STRING
                         0000V 31   000B4          BRW     169$
                00000000G    EF   9F 000B7 2$:      PUSHAB  SHIFT                                   ; 0479
                             04   DD 000BD          PUSHL   #4
                00000000G    EF   9F 000BF          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 000C5          CALLS   #3,PAS$WRITE_STRING
                FFFFBFC0     EF   9F 000CC          PUSHAB  C.ACA
                             21   DD 000D2          PUSHL   #33
                00000000G    EF   9F 000D4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 000DA          CALLS   #3,PAS$WRITE_STRING
                00000014G    EF   9F 000E1          PUSHAB  IDATA+20
00000000G  EF                01   FB 000E7          CALLS   #1,NUM_LEN
                             50   DD 000EE          PUSHL   R0
                00000014G    EF   DD 000F0          PUSHL   IDATA+20
                00000000G    EF   9F 000F6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 000FC          CALLS   #3,PAS$WRITE_INTEGER
                FFFFBFAD     EF   9F 00103          PUSHAB  C.ACB
                             08   DD 00109          PUSHL   #8
                00000000G    EF   9F 0010B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 00111          CALLS   #3,PAS$WRITE_STRING
                00000000G    EF   9F 00118          PUSHAB  DEF
00000000G  EF                01   FB 0011E          CALLS   #1,NUM_LEN
                             50   DD 00125          PUSHL   R0
                00000000G    EF   DD 00127          PUSHL   DEF
                00000000G    EF   9F 0012D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 00133          CALLS   #3,PAS$WRITE_INTEGER
                             01   DD 0013A          PUSHL   #1
             7E        5D    8F   9A 0013C          MOVZBL  #93,-(SP)
                00000000G    EF   9F 00140          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 00146          CALLS   #3,PAS$WRITE_CHAR
                00000014G    EF   9F 0014D          PUSHAB  IDATA+20                                ; 0483
00000000G  EF                01   FB 00153          CALLS   #1,NUM_LEN
             5C              50   D0 0015A          MOVL    R0,R12
                00000000G    EF   9F 0015D          PUSHAB  DEF
00000000G  EF                01   FB 00163          CALLS   #1,NUM_LEN
             50          604C  9E   0016A          MOVAB   (R0)[R12],R0
             03            50  D1   0016E          CMPL    R0,#3
                           00V 14   00171          BGTR    4$
                FFFFBF45     EF   9F 00173          PUSHAB  C.ACC                                   ; 0485
                             03   DD 00179          PUSHL   #3
                00000000G    EF   9F 0017B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 00181          CALLS   #3,PAS$WRITE_STRING
                         0000V 31   00188          BRW     169$
                FFFFBF31     EF   9F 0018B 4$:      PUSHAB  C.ACD                                   ; 0489
                             03   DD 00191          PUSHL   #3
                00000000G    EF   9F 00193          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 00199          CALLS   #3,PAS$WRITE_STRING
                         0000V 31   001A0          BRW     169$
                00000000G    EF   9F 001A3 6$:      PUSHAB  SHIFT                                   ; 0497
                             04   DD 001A9          PUSHL   #4
                00000000G    EF   9F 001AB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03   FB 001B1          CALLS   #3,PAS$WRITE_STRING
                FFFFBF08     EF   9F 001B8          PUSHAB  C.ACE
```

```
                                 2F    DD 001BE            PUSHL   #47
                      00000000G  EF    9F 001C0            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 001C6            CALLS   #3,PAS$WRITE_STRING
                               0000V   31 001CD            BRW     169$
                      00000000G  EF    9F 001D0 7$:        PUSHAB  SHIFT                                   ; 0506
                                 04    DD 001D6            PUSHL   #4
                      00000000G  EF    9F 001D8            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 001DE            CALLS   #3,PAS$WRITE_STRING
                      FFFFBF0B   EF    9F 001E5            PUSHAB  C.ACF
                                 21    DD 001EB            PUSHL   #33
                      00000000G  EF    9F 001ED            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 001F3            CALLS   #3,PAS$WRITE_STRING
                      00000014G  EF    9F 001FA            PUSHAB  IDATA+20
00000000G  EF                    01    FB 00200            CALLS   #1,NUM_LEN
                                 50    DD 00207            PUSHL   R0
                      00000014G  EF    DD 00209            PUSHL   IDATA+20
                      00000000G  EF    9F 0020F            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 00215            CALLS   #3,PAS$WRITE_INTEGER
                      FFFFBEF8   EF    9F 0021C            PUSHAB  C.ACG
                                 08    DD 00222            PUSHL   #8
                      00000000G  EF    9F 00224            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0022A            CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF    9F 00231            PUSHAB  DEF
00000000G  EF                    01    FB 00237            CALLS   #1,NUM_LEN
                                 50    DD 0023E            PUSHL   R0
                      00000000G  EF    DD 00240            PUSHL   DEF
                      00000000G  EF    9F 00246            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0024C            CALLS   #3,PAS$WRITE_INTEGER
                                 01    DD 00253            PUSHL   #1
               7E        5D      8F    9A 00255            MOVZBL  #93,-(SP)
                      00000000G  EF    9F 00259            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0025F            CALLS   #3,PAS$WRITE_CHAR
                      00000014G  EF    9F 00266            PUSHAB  IDATA+20                                ; 0510
00000000G  EF                    01    FB 0026C            CALLS   #1,NUM_LEN
               5C                50    D0 00273            MOVL    R0,R12
                      00000000G  EF    9F 00276            PUSHAB  DEF
00000000G  EF                    01    FB 0027C            CALLS   #1,NUM_LEN
               50              604C    9E 00283            MOVAB   (R0)[R12],R0
               03                50    D1 00287            CMPL    R0,#3
                                 00V   14 0028A            BGTR    9$
                      FFFFBE90   EF    9F 0028C            PUSHAB  C.ACH                                   ; 0512
                                 03    DD 00292            PUSHL   #3
                      00000000G  EF    9F 00294            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0029A            CALLS   #3,PAS$WRITE_STRING
                               0000V   31 002A1            BRW     169$
                      FFFFBE7C   EF    9F 002A4 9$:        PUSHAB  C.ACI                                   ; 0516
                                 03    DD 002AA            PUSHL   #3
                      00000000G  EF    9F 002AC            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 002B2            CALLS   #3,PAS$WRITE_STRING
                               0000V   31 002B9            BRW     169$
                      00000000G  EF    9F 002BC 11$:       PUSHAB  SHIFT                                   ; 0522
                                 04    DD 002C2            PUSHL   #4
                      00000000G  EF    9F 002C4            PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 002CA            CALLS   #3,PAS$WRITE_STRING
                      FFFFBE53   EF    9F 002D1            PUSHAB  C.ACJ
                                 0E    DD 002D7            PUSHL   #14
                      00000000G  EF    9F 002D9            PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF                03  FB 002DF          CALLS   #3,PAS$WRITE_STRING
                             03  DD 002E6          PUSHL   #3
                   00000084G EF  DD 002E8          PUSHL   IDATA+132
                   00000000G EF  9F 002EE          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 002F4          CALLS   #3,PAS$WRITE_INTEGER
                   FFFFBE39  EF  9F 002FB          PUSHAB  C.ACK
                             0B  DD 00301          PUSHL   #11
                   00000000G EF  9F 00303          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00309          CALLS   #3,PAS$WRITE_STRING
                   00000000G EF  9F 00310          PUSHAB  MAX_KEY_SIZE
00000000G  EF                01  FB 00316          CALLS   #1,NUM_LEN
                             50  DD 0031D          PUSHL   R0
                   00000000G EF  DD 0031F          PUSHL   MAX_KEY_SIZE
                   00000000G EF  9F 00325          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0032B          CALLS   #3,PAS$WRITE_INTEGER
                   FFFFBE0E  EF  9F 00332          PUSHAB  C.ACL
                             07  DD 00338          PUSHL   #7
                   00000000G EF  9F 0033A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00340          CALLS   #3,PAS$WRITE_STRING
                        0000V 31  00347          BRW     169$
                   00000000G EF  9F 0034A 12$:     PUSHAB  SHIFT                                   ; 0529
                             04  DD 00350          PUSHL   #4
                   00000000G EF  9F 00352          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00358          CALLS   #3,PAS$WRITE_STRING
                   FFFFBDE9  EF  9F 0035F          PUSHAB  C.ACM
                             0F  DD 00365          PUSHL   #15
                   00000000G EF  9F 00367          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0036D          CALLS   #3,PAS$WRITE_STRING
                             03  DD 00374          PUSHL   #3
                   00000084G EF  DD 00376          PUSHL   IDATA+132
                   00000000G EF  9F 0037C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00382          CALLS   #3,PAS$WRITE_INTEGER
                   FFFFBDCF  EF  9F 00389          PUSHAB  C.ACN
                             09  DD 0038F          PUSHL   #9
                   00000000G EF  9F 00391          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00397          CALLS   #3,PAS$WRITE_STRING
                   00000014G EF  9F 0039E          PUSHAB  IDATA+20
00000000G  EF                01  FB 003A4          CALLS   #1,NUM_LEN
                             50  DD 003AB          PUSHL   R0
                   00000014G EF  DD 003AD          PUSHL   IDATA+20
                   00000000G EF  9F 003B3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 003B9          CALLS   #3,PAS$WRITE_INTEGER
                             01  DD 003C0          PUSHL   #1
                             2D  DD 003C2          PUSHL   #45
                   00000000G EF  9F 003C4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 003CA          CALLS   #3,PAS$WRITE_CHAR
                   00000000G EF  9F 003D1          PUSHAB  MAX_KEY_SIZE
00000000G  EF                01  FB 003D7          CALLS   #1,NUM_LEN
                             50  DD 003DE          PUSHL   R0
                   00000000G EF  DD 003E0          PUSHL   MAX_KEY_SIZE
                   00000000G EF  9F 003E6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 003EC          CALLS   #3,PAS$WRITE_INTEGER
                   FFFFBD6F  EF  9F 003F3          PUSHAB  C.ACO
                             02  DD 003F9          PUSHL   #2
                   00000000G EF  9F 003FB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00401          CALLS   #3,PAS$WRITE_STRING
                   00000000G EF  9F 00408          PUSHAB  MAX_KEY_SIZE
```

Generated Code

```
00000000G  EF              01   FB 0040E          CALLS   #1,NUM_LEN
                           50   DD 00415          PUSHL   R0
                 00000000G EF   DD 00417          PUSHL   MAX_KEY_SIZE
                 00000000G EF   9F 0041D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00423          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFBD3A  EF   9F 0042A          PUSHAB  C.ACP
                           04   DD 00430          PUSHL   #4
                 00000000G EF   9F 00432          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00438          CALLS   #3,PAS$WRITE_STRING
                       0000V 31   0043F          BRW     169$
                 00000000G EF   9F 00442 13$:     PUSHAB  SHIFT                                   ; 0539
                           04   DD 00448          PUSHL   #4
                 00000000G EF   9F 0044A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00450          CALLS   #3,PAS$WRITE_STRING
                 FFFFBD11  EF   9F 00457          PUSHAB  C.ACQ
                           1B   DD 0045D          PUSHL   #27
                 00000000G EF   9F 0045F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00465          CALLS   #3,PAS$WRITE_STRING
                           05   DD 0046C          PUSHL   #5
                 00000000G EF   DD 0046E          PUSHL   CUR_MAX_REC
                 00000000G EF   9F 00474          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0047A          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFBD03  EF   9F 00481          PUSHAB  C.ACR
                           07   DD 00487          PUSHL   #7
                 00000000G EF   9F 00489          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0048F          CALLS   #3,PAS$WRITE_STRING
                       0000V 31   00496          BRW     169$
                 00000000G EF   9F 00499 14$:     PUSHAB  SHIFT                                   ; 0548
                           04   DD 0049F          PUSHL   #4
                 00000000G EF   9F 004A1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004A7          CALLS   #3,PAS$WRITE_STRING
                 FFFFBCDE  EF   9F 004AE          PUSHAB  C.ACS
                           1A   DD 004B4          PUSHL   #26
                 00000000G EF   9F 004B6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004BC          CALLS   #3,PAS$WRITE_STRING
                 00000014G EF   9F 004C3          PUSHAB  IDATA+20
00000000G  EF              01   FB 004C9          CALLS   #1,NUM_LEN
                           50   DD 004D0          PUSHL   R0
                 00000014G EF   DD 004D2          PUSHL   IDATA+20
                 00000000G EF   9F 004D8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004DE          CALLS   #3,PAS$WRITE_INTEGER
                           01   DD 004E5          PUSHL   #1
                           2D   DD 004E7          PUSHL   #45
                 00000000G EF   9F 004E9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 004EF          CALLS   #3,PAS$WRITE_CHAR
                           05   DD 004F6          PUSHL   #5
                 00000000G EF   DD 004F8          PUSHL   CUR_MAX_REC
                 00000000G EF   9F 004FE          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00504          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFBC9D  EF   9F 0050B          PUSHAB  C.ACT
                           07   DD 00511          PUSHL   #7
                 00000000G EF   9F 00513          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00519          CALLS   #3,PAS$WRITE_STRING
                 00000014G EF   9F 00520          PUSHAB  IDATA+20                                ; 0552
00000000G  EF              01   FB 00526          CALLS   #1,NUM_LEN
           03              50   D1 0052D          CMPL    R0,#3
                           00V  18 00530          BGEQ    16$
```

```
                FFFFBC7E   EF   9F 00532          PUSHAB  C.ACU                                   ; 0554
                           03   DD 00538          PUSHL   #3
                00000000G  EF   9F 0053A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00540          CALLS   #3,PAS$WRITE_STRING
                      0000V 31   00547          BRW     169$
                FFFFBC6A   EF   9F 0054A 16$:     PUSHAB  C.ACV                                   ; 0558
                           03   DD 00550          PUSHL   #3
                00000000G  EF   9F 00552          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00558          CALLS   #3,PAS$WRITE_STRING
                      0000V 31   0055F          BRW     169$
                00000000G  EF   9F 00562 18$:     PUSHAB  SHIFT                                   ; 0564
                           04   DD 00568          PUSHL   #4
                00000000G  EF   9F 0056A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00570          CALLS   #3,PAS$WRITE_STRING
                FFFFBC41   EF   9F 00577          PUSHAB  C.ACW
                           0E   DD 0057D          PUSHL   #14
                00000000G  EF   9F 0057F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00585          CALLS   #3,PAS$WRITE_STRING
                           03   DD 0058C          PUSHL   #3
                00000084G  EF   DD 0058E          PUSHL   IDATA+132
                00000000G  EF   9F 00594          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0059A          CALLS   #3,PAS$WRITE_INTEGER
                FFFFBC27   EF   9F 005A1          PUSHAB  C.ACX
                           1C   DD 005A7          PUSHL   #28
                00000000G  EF   9F 005A9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 005AF          CALLS   #3,PAS$WRITE_STRING
                      0000V 31   005B6          BRW     169$
                00000000G  EF   9F 005B9 19$:     PUSHAB  SHIFT                                   ; 0570
                           04   DD 005BF          PUSHL   #4
                00000000G  EF   9F 005C1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 005C7          CALLS   #3,PAS$WRITE_STRING
                FFFFBC16   EF   9F 005CE          PUSHAB  C.ACY
                           0F   DD 005D4          PUSHL   #15
                00000000G  EF   9F 005D6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 005DC          CALLS   #3,PAS$WRITE_STRING
                           03   DD 005E3          PUSHL   #3
                00000084G  EF   DD 005E5          PUSHL   IDATA+132
                00000000G  EF   9F 005EB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 005F1          CALLS   #3,PAS$WRITE_INTEGER
                FFFFBBFC   EF   9F 005F8          PUSHAB  C.ACZ
                           1D   DD 005FE          PUSHL   #29
                00000000G  EF   9F 00600          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00606          CALLS   #3,PAS$WRITE_STRING
                      0000V 31   0060D          BRW     169$
                00000000   8F   DF 00610 20$:     PUSHAL  #0                                      ; 0578
00000000G  EF              01   FB 00616          CALLS   #1,CLEAR
00V00000000G  EF           00   E0 0061D          BBS     #0,FULL_PROMPT,22$                      ; 0583
03 00000000G  EF           00   E0 00625          BBS     #0,TEMP_FULL_PROMPT,.+3
                      0000V 31   0062D          BRW     23$
                00000000G  EF   9F 00630 22$:     PUSHAB  SHIFT                                   ; 0590
                           04   DD 00636          PUSHL   #4
                00000000G  EF   9F 00638          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0063E          CALLS   #3,PAS$WRITE_STRING
                FFFFBBCD   EF   9F 00645          PUSHAB  C.ADA
                           02   DD 0064B          PUSHL   #2
                00000000G  EF   9F 0064D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 00653          CALLS   #3,PAS$WRITE_STRING
```

```
                                Generated Code

                                00000000G  EF   9F 0065A          PUSHAB  ANSI_REVERSE
                                           04   DD 00660          PUSHL   #4
                                00000000G  EF   9F 00662          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00668          CALLS   #3,PAS$WRITE_STRING
                                FFFFBBA5   EF   9F 0066F          PUSHAB  C.ADB
                                           18   DD 00675          PUSHL   #24
                                00000000G  EF   9F 00677          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0067D          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00684          PUSHAB  ANSI_RESET
                                           04   DD 0068A          PUSHL   #4
                                00000000G  EF   9F 0068C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00692          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00699          PUSHAB  CRLF
                                           02   DD 0069F          PUSHL   #2
                                00000000G  EF   9F 006A1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 006A7          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 006AE          PUSHAB  CRLF_SHIFT
                                           06   DD 006B4          PUSHL   #6
                                00000000G  EF   9F 006B6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 006BC          CALLS   #3,PAS$WRITE_STRING
                                FFFFBB69   EF   9F 006C3          PUSHAB  C.ADC
                                           3A   DD 006C9          PUSHL   #58
                                00000000G  EF   9F 006CB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 006D1          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 006D8          PUSHAB  CRLF_SHIFT
                                           06   DD 006DE          PUSHL   #6
                                00000000G  EF   9F 006E0          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 006E6          CALLS   #3,PAS$WRITE_STRING
                                FFFFBB7B   EF   9F 006ED          PUSHAB  C.ADD
                                           34   DD 006F3          PUSHL   #52
                                00000000G  EF   9F 006F5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 006FB          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00702          PUSHAB  CRLF_SHIFT
                                           06   DD 00708          PUSHL   #6
                                00000000G  EF   9F 0070A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00710          CALLS   #3,PAS$WRITE_STRING
                                FFFFBB85   EF   9F 00717          PUSHAB  C.ADE
                                           3A   DD 0071D          PUSHL   #58
                                00000000G  EF   9F 0071F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00725          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 0072C          PUSHAB  CRLF_SHIFT
                                           06   DD 00732          PUSHL   #6
                                00000000G  EF   9F 00734          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0073A          CALLS   #3,PAS$WRITE_STRING
                                FFFFBB97   EF   9F 00741          PUSHAB  C.ADF
                                00000040   8F   DD 00747          PUSHL   #64
                                00000000G  EF   9F 0074D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00753          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 0075A          PUSHAB  CRLF_SHIFT
                                           06   DD 00760          PUSHL   #6
                                00000000G  EF   9F 00762          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 00768          CALLS   #3,PAS$WRITE_STRING
                                FFFFBBA9   EF   9F 0076F          PUSHAB  C.ADG
                                           34   DD 00775          PUSHL   #52
                                00000000G  EF   9F 00777          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0077D          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 00784          PUSHAB  CRLF_SHIFT
```

```
                                 06    DD 0078A          PUSHL   #6
                    00000000G    EF    9F 0078C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00792          CALLS   #3,PAS$WRITE_STRING
                    FFFFBBB3     EF    9F 00799          PUSHAB  C.ADH
                                 38    DD 0079F          PUSHL   #56
                    00000000G    EF    9F 007A1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 007A7          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 007AE          PUSHAB  CRLF_SHIFT
                                 06    DD 007B4          PUSHL   #6
                    00000000G    EF    9F 007B6          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 007BC          CALLS   #3,PAS$WRITE_STRING
                    FFFFBBC1     EF    9F 007C3          PUSHAB  C.ADI
                                 38    DD 007C9          PUSHL   #56
                    00000000G    EF    9F 007CB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 007D1          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 007D8          PUSHAB  CRLF
                                 02    DD 007DE          PUSHL   #2
                    00000000G    EF    9F 007E0          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 007E6          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 007ED          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01    FB 007F3          CALLS   #1,PAS$WRITELN2
                                 00V   11 007FA          BRB     24$
                    00000000G    EF    9F 007FC 23$:     PUSHAB  SHIFT                                   ; 0618
                                 04    DD 00802          PUSHL   #4
                    00000000G    EF    9F 00804          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0080A          CALLS   #3,PAS$WRITE_STRING
                    FFFFBBAB     EF    9F 00811          PUSHAB  C.ADJ
                                 24    DD 00817          PUSHL   #36
                    00000000G    EF    9F 00819          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0081F          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 00826          PUSHAB  CRLF_SHIFT
                                 06    DD 0082C          PUSHL   #6
                    00000000G    EF    9F 0082E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00834          CALLS   #3,PAS$WRITE_STRING
                    FFFFBBA5     EF    9F 0083B          PUSHAB  C.ADK
                                 1D    DD 00841          PUSHL   #29
                    00000000G    EF    9F 00843          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00849          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 00850          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01    FB 00856          CALLS   #1,PAS$WRITELN2
                    00000000G    EF    9F 0085D 24$:     PUSHAB  SHIFT                                   ; 0626
                                 04    DD 00863          PUSHL   #4
                    00000000G    EF    9F 00865          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 0086B          CALLS   #3,PAS$WRITE_STRING
                    FFFFBB8E     EF    9F 00872          PUSHAB  C.ADL
                                 1F    DD 00878          PUSHL   #31
                    00000000G    EF    9F 0087A          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00880          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 00887          PUSHAB  ANSI_REVERSE
                                 04    DD 0088D          PUSHL   #4
                    00000000G    EF    9F 0088F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 00895          CALLS   #3,PAS$WRITE_STRING
                    FFFFBB84     EF    9F 0089C          PUSHAB  C.ADM
                                 03    DD 008A2          PUSHL   #3
                    00000000G    EF    9F 008A4          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03    FB 008AA          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 008B1          PUSHAB  ANSI_RESET
```

```
                                 04    DD 008B7          PUSHL   #4
                      00000000G  EF    9F 008B9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 008BF          CALLS   #3,PAS$WRITE_STRING
                      FFFFBB5E   EF    9F 008C6          PUSHAB  C.ADN
                                 03    DD 008CC          PUSHL   #3
                      00000000G  EF    9F 008CE          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 008D4          CALLS   #3,PAS$WRITE_STRING
                               0000V   31 008DB          BRW     169$
                      00000000G  EF    9F 008DE 25$:     PUSHAB  CRLF_SHIFT                              ; 0633
                                 06    DD 008E4          PUSHL   #6
                      00000000G  EF    9F 008E6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 008EC          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF    9F 008F3          PUSHAB  ANSI_REVERSE
                                 04    DD 008F9          PUSHL   #4
                      00000000G  EF    9F 008FB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 00901          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF    9F 00908          PUSHAB  CONTINUE_TEXT
                                 2D    DD 0090E          PUSHL   #45
                      00000000G  EF    9F 00910          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 00916          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF    9F 0091D          PUSHAB  ANSI_RESET
                                 04    DD 00923          PUSHL   #4
                      00000000G  EF    9F 00925          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0092B          CALLS   #3,PAS$WRITE_STRING
                                 01    DD 00932          PUSHL   #1
                                 09    DD 00934          PUSHL   #9
                      00000000G  EF    9F 00936          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03    FB 0093C          CALLS   #3,PAS$WRITE_CHAR
                               0000V   31 00943          BRW     169$
                      00000000G  EF    9F 00946 26$:     PUSHAB  SHIFT                                   ; 0638
                                 04    DD 0094C          PUSHL   #4
                      00000000G  EF    9F 0094E          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                 03    FB 00954          CALLS   #3,PAS$WRITE_STRING
                      FFFFBACD   EF    9F 0095B          PUSHAB  C.ADO
                                 1C    DD 00961          PUSHL   #28
                      00000000G  EF    9F 00963          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                 03    FB 00969          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF    9F 00970          PUSHAB  CRLF_SHIFT
                                 06    DD 00976          PUSHL   #6
                      00000000G  EF    9F 00978          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                 03    FB 0097E          CALLS   #3,PAS$WRITE_STRING
                      FFFFBABF   EF    9F 00985          PUSHAB  C.ADP
                                 2B    DD 0098B          PUSHL   #43
                      00000000G  EF    9F 0098D          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                 03    FB 00993          CALLS   #3,PAS$WRITE_STRING
                               0000V   31 0099A          BRW     169$
                      00000003   8F    DF 0099D 27$:     PUSHAL  #3                                      ; 0646
   00000000G  EF                 01    FB 009A3          CALLS   #1,CLEAR
00V00000000G  EF                 00    E0 009AA          BBS     #0,FULL_PROMPT,29$                      ; 0651
03 00000000G  EF                 00    E0 009B2          BBS     #0,TEMP_FULL_PROMPT,..+3
                               0000V   31 009BA          BRW     30$
                      00000000G  EF    9F 009BD 29$:     PUSHAB  SHIFT                                   ; 0659
                                 04    DD 009C3          PUSHL   #4
                      00000000G  EF    9F 009C5          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                 03    FB 009CB          CALLS   #3,PAS$WRITE_STRING
                      FFFFBA9E   EF    9F 009D2          PUSHAB  C.ADQ
                                 02    DD 009D8          PUSHL   #2
```

```
                 00000000G  EF  9F 009DA         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 009E0         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 009E7         PUSHAB  ANSI_REVERSE
                            04  DD 009ED         PUSHL   #4
                 00000000G  EF  9F 009EF         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 009F5         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 009FC         PUSHAB  EDF_HEADER
                            13  DD 00A02         PUSHL   #19
                 00000000G  EF  9F 00A04         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A0A         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00A11         PUSHAB  ANSI_RESET
                            04  DD 00A17         PUSHL   #4
                 00000000G  EF  9F 00A19         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A1F         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00A26         PUSHAB  CRLF
                            02  DD 00A2C         PUSHL   #2
                 00000000G  EF  9F 00A2E         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A34         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00A3B         PUSHAB  CRLF_SHIFT
                            06  DD 00A41         PUSHL   #6
                 00000000G  EF  9F 00A43         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A49         CALLS   #3,PAS$WRITE_STRING
                 FFFFBA24   EF  9F 00A50         PUSHAB  C.ADR
                            37  DD 00A56         PUSHL   #55
                 00000000G  EF  9F 00A58         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A5E         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00A65         PUSHAB  CRLF_SHIFT
                            06  DD 00A6B         PUSHL   #6
                 00000000G  EF  9F 00A6D         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A73         CALLS   #3,PAS$WRITE_STRING
                 FFFFBA32   EF  9F 00A7A         PUSHAB  C.ADS
                            3A  DD 00A80         PUSHL   #58
                 00000000G  EF  9F 00A82         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A88         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00A8F         PUSHAB  CRLF_SHIFT
                            06  DD 00A95         PUSHL   #6
                 00000000G  EF  9F 00A97         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00A9D         CALLS   #3,PAS$WRITE_STRING
                 FFFFBA44   EF  9F 00AA4         PUSHAB  C.ADT
                            38  DD 00AAA         PUSHL   #56
                 00000000G  EF  9F 00AAC         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00AB2         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00AB9         PUSHAB  CRLF_SHIFT
                            06  DD 00ABF         PUSHL   #6
                 00000000G  EF  9F 00AC1         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00AC7         CALLS   #3,PAS$WRITE_STRING
                 FFFFBA52   EF  9F 00ACE         PUSHAB  C.ADU
                            2F  DD 00AD4         PUSHL   #47
                 00000000G  EF  9F 00AD6         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00ADC         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 00AE3         PUSHAB  CRLF_SHIFT
                            06  DD 00AE9         PUSHL   #6
                 00000000G  EF  9F 00AEB         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB 00AF1         CALLS   #3,PAS$WRITE_STRING
                 FFFFBA58   EF  9F 00AF8         PUSHAB  C.ADV
                            30  DD 00AFE         PUSHL   #48
                 00000000G  EF  9F 00B00         PUSHAB  PAS$FV_OUTPUT
```

Generated Code

```
00000000G  EF                03  FB 00B06          CALLS   #3,PAS$WRITE_STRING
                  00000000G  EF  9F 00B0D          PUSHAB  CRLF_SHIFT
                             06  DD 00B13          PUSHL   #6
                  00000000G  EF  9F 00B15          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B1B          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA5E   EF  9F 00B22          PUSHAB  C.ADW
                             37  DD 00B28          PUSHL   #55
                  00000000G  EF  9F 00B2A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B30          CALLS   #3,PAS$WRITE_STRING
                  00C00000G  EF  9F 00B37          PUSHAB  CRLF_SHIFT
                             06  DD 00B3D          PUSHL   #6
                  00000000G  EF  9F 00B3F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B45          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA6C   EF  9F 00B4C          PUSHAB  C.ADX
                             36  DD 00B52          PUSHL   #54
                  00000000G  EF  9F 00B54          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B5A          CALLS   #3,PAS$WRITE_STRING
                  00000000G  EF  9F 00B61          PUSHAB  CRLF_SHIFT
                             06  DD 00B67          PUSHL   #6
                  00000000G  EF  9F 00B69          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B6F          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA7A   EF  9F 00B76          PUSHAB  C.ADY
                             29  DD 00B7C          PUSHL   #41
                  00000000G  EF  9F 00B7E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B84          CALLS   #3,PAS$WRITE_STRING
                  00000000G  EF  9F 00B8B          PUSHAB  CRLF_SHIFT
                             06  DD 00B91          PUSHL   #6
                  00000000G  EF  9F 00B93          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00B99          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA7C   EF  9F 00BA0          PUSHAB  C.ADZ
                             2A  DD 00BA6          PUSHL   #42
                  00000000G  EF  9F 00BA8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00BAE          CALLS   #3,PAS$WRITE_STRING
                  00000000G  EF  9F 00BB5          PUSHAB  CRLF
                             02  DD 00BBB          PUSHL   #2
                  00000000G  EF  9F 00BBD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00BC3          CALLS   #3,PAS$WRITE_STRING
                  00000000G  EF  9F 00BCA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                01  FB 00BD0          CALLS   #1,PAS$WRITELN2
                             00V 11  00BD7          BRB     31$
                  00000000G  EF  9F 00BD9 30$:     PUSHAB  SHIFT                           ; 0690
                             04  DD 00BDF          PUSHL   #4
                  00000000G  EF  9F 00BE1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00BE7          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA5A   EF  9F 00BEE          PUSHAB  C.AEA
                             32  DD 00BF4          PUSHL   #50
                  00000000G  EF  9F 00BF6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00BFC          CALLS   #3 PAS$WRITE_STRING
                  00000000G  EF  9F 00C03          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                01  FB 00C09          CALLS   #1,PAS$WRITELN2
                  00000000G  EF  9F 00C10 31$:     PUSHAB  SHIFT                           ; 0696
                             04  DD 00C16          PUSHL   #4
                  00000000G  EF  9F 00C18          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 00C1E          CALLS   #3,PAS$WRITE_STRING
                  FFFFBA57   EF  9F 00C25          PUSHAB  C.AEB
                             28  DD 00C2B          PUSHL   #40
                  00000000G  EF  9F 00C2D          PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF  03  FB 00C33          CALLS   #3,PAS$WRITE_STRING
          0000V  31 00C3A            BRW     169$
               00000000G  EF  9F 00C3D 32$:  PUSHAB  SHIFT                          ; 0706
                          04  DD 00C43       PUSHL   #4
               00000000G  EF  9F 00C45       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00C4B       CALLS   #3,PAS$WRITE_STRING
               FFFFBA52   EF  9F 00C52       PUSHAB  C.AEC
                          12  DD 00C58       PUSHL   #18
               00000000G  EF  9F 00C5A       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00C60       CALLS   #3,PAS$WRITE_STRING
               00000000G  EF  9F 00C67       PUSHAB  CRLF_SHIFT
                          06  DD 00C6D       PUSHL   #6
               00000000G  EF  9F 00C6F       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00C75       CALLS   #3,PAS$WRITE_STRING
               FFFFBA3C   EF  9F 00C7C       PUSHAB  C.AED
                          2F  DD 00C82       PUSHL   #47
               00000000G  EF  9F 00C84       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00C8A       CALLS   #3,PAS$WRITE_STRING
          0000V  31 00C91            BRW     169$
               00000000G  EF  9F 00C94 33$:  PUSHAB  SHIFT                          ; 0715
                          04  DD 00C9A       PUSHL   #4
               00000000G  EF  9F 00C9C       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00CA2       CALLS   #3,PAS$WRITE_STRING
               FFFFBA3F   EF  9F 00CA9       PUSHAB  C.AEE
                          0C  DD 00CAF       PUSHL   #12
               00000000G  EF  9F 00CB1       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00CB7       CALLS   #3,PAS$WRITE_STRING
               00000000G  EF  9F 00CBE       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             01  FB 00CC4       CALLS   #1,PAS$WRITELN2
               00000000G  EF  9F 00CCB       PUSHAB  SHIFT                          ; 0716
                          04  DD 00CD1       PUSHL   #4
               00000000G  EF  9F 00CD3       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00CD9       CALLS   #3,PAS$WRITE_STRING
               FFFFBA14   EF  9F 00CE0       PUSHAB  C.AEF
                          2C  DD 00CE6       PUSHL   #44
               00000000G  EF  9F 00CE8       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00CEE       CALLS   #3,PAS$WRITE_STRING
          0000V  31 00CF5            BRW     169$
               00000000G  EF  9F 00CF8 34$:  PUSHAB  SHIFT                          ; 0725
                          04  DD 00CFE       PUSHL   #4
               00000000G  EF  9F 00D00       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00D06       CALLS   #3,PAS$WRITE_STRING
               FFFFBA13   EF  9F 00D0D       PUSHAB  C.AEG
                          03  DD 00D13       PUSHL   #3
               00000000G  EF  9F 00D15       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00D1B       CALLS   #3,PAS$WRITE_STRING
                          03  DD 00D22       PUSHL   #3
               00000084G  EF  DD 00D24       PUSHL   IDATA+132
               00000000G  EF  9F 00D2A       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00D30       CALLS   #3,PAS$WRITE_INTEGER
               FFFFB9ED   EF  9F 00D37       PUSHAB  C.AEH
                          0B  DD 00D3D       PUSHL   #11
               00000000G  EF  9F 00D3F       PUSHAB  PAS$FV_OUTPUT
00000000G  EF             03  FB 00D45       CALLS   #3,PAS$WRITE_STRING
00V00000013G  EF          00  E1 00D4C       BBC     #0,BDATA+19,36$                ; 0728
               FFFFB9DC   EF  9F 00D54       PUSHAB  C.AEI                          ; 0730
                          03  DD 00D5A       PUSHL   #3
```

```
                    00000000G   EF   9F 00D5C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00D62          CALLS   #3,PAS$WRITE_STRING
                                01   DD 00D69          PUSHL   #1
                    00000000G   EF   DD 00D6B          PUSHL   SEGMENT_NUMBER
                    00000000G   EF   9F 00D71          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00D77          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB9B6    EF   9F 00D7E 36$:     PUSHAB  C.AEJ                                   ; 0732
                                04   DD 00D84          PUSHL   #4
                    00000000G   EF   9F 00D86          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00D8C          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 00D93          PUSHAB  MAX_KEY_POSITION
          00000000G EF          01   FB 00D99          CALLS   #1,NUM_LEN
                                50   DD 00DA0          PUSHL   R0
                    00000000G   EF   DD 00DA2          PUSHL   MAX_KEY_POSITION
                    00000000G   EF   9F 00DA8          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00DAE          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB983    EF   9F 00DB5          PUSHAB  C.AEK
                                07   DD 00DBB          PUSHL   #7
                    00000000G   EF   9F 00DBD          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00DC3          CALLS   #3,PAS$WRITE_STRING
                              0000V  31   00DCA          BRW     169$
                    00000000G   EF   9F 00DCD 37$:     PUSHAB  SHIFT                                   ; 0741
                                04   DD 00DD3          PUSHL   #4
                    00000000G   EF   9F 00DD5          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00DDB          CALLS   #3,PAS$WRITE_STRING
                    FFFFB95E    EF   9F 00DE2          PUSHAB  C.AEL
                                31   DD 00DE8          PUSHL   #49
                    00000000G   EF   9F 00DEA          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00DF0          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 00DF7          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          01   FB 00DFD          CALLS   #1,PAS$WRITELN2
       00V00000000G EF          00   E0 00E04          BBS     #0,OPTIMIZING,39$                       ; 0744
                    00000000G   EF   9F 00E0C          PUSHAB  SHIFT                                   ; 0746
                                04   DD 00E12          PUSHL   #4
                    00000000G   EF   9F 00E14          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00E1A          CALLS   #3,PAS$WRITE_STRING
                    FFFFB953    EF   9F 00E21          PUSHAB  C.AEM
                                07   DD 00E27          PUSHL   #7
                    00000000G   EF   9F 00E29          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00E2F          CALLS   #3,PAS$WRITE_STRING
                                00V  11   00E36          BRB     40$
                    00000000G   EF   9F 00E38 39$:     PUSHAB  SHIFT                                   ; 0750
                                04   DD 00E3E          PUSHL   #4
                    00000000G   EF   9F 00E40          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00E46          CALLS   #3,PAS$WRITE_STRING
                    FFFFB92F    EF   9F 00E4D          PUSHAB  C.AEN
                                08   DD 00E53          PUSHL   #8
                    00000000G   EF   9F 00E55          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00E5B          CALLS   #3,PAS$WRITE_STRING
                    FFFFB922    EF   9F 00E62 40$:     PUSHAB  C.AEO                                   ; 0752
                                18   DD 00E68          PUSHL   #24
                    00000000G   EF   9F 00E6A          PUSHAB  PAS$FV_OUTPUT
          00000000G EF          03   FB 00E70          CALLS   #3,PAS$WRITE_STRING
       00V00000000G EF          00   E0 00E77          BBS     #0,OPTIMIZING,42$                       ; 0754
                                01   DD 00E7F          PUSHL   #1                                      ; 0756
                 7E 00000000G   EF   9A 00E81          MOVZBL  TAB,-(SP)
                    00000000G   EF   9F 00E88          PUSHAB  PAS$FV_OUTPUT
```

```
                00000000G  EF                 03  FB 00E8E                CALLS   #3,PAS$WRITE_CHAR
                           FFFFB907           EF  9F 00E95 42$:           PUSHAB  C.AEP                       ; 0758
                                              0F  DD 00E9B                PUSHL   #15
                           00000000G          EF  9F 00E9D                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00EA3                CALLS   #3,PAS$WRITE_STRING
                                         0000V  31  00EAA                BRW     169$
                           00000000G          EF  9F 00EAD 43$:           PUSHAB  SHIFT                       ; 0764
                                              04  DD 00EB3                PUSHL   #4
                           00000000G          EF  9F 00EB5                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00EBB                CALLS   #3,PAS$WRITE_STRING
                           FFFFB8EA           EF  9F 00EC2                PUSHAB  C.AEQ
                                              03  DD 00EC8                PUSHL   #3
                           00000000G          EF  9F 00ECA                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00ED0                CALLS   #3,PAS$WRITE_STRING
                                              03  DD 00ED7                PUSHL   #3
                           00000084G          EF  DD 00ED9                PUSHL   IDATA+132
                           00000000G          EF  9F 00EDF                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00EE5                CALLS   #3,PAS$WRITE_INTEGER
                           FFFFB8C4           EF  9F 00EEC                PUSHAB  C.AER
                                              22  DD 00EF2                PUSHL   #34
                           00000000G          EF  9F 00EF4                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00EFA                CALLS   #3,PAS$WRITE_STRING
                                         0000V  31  00F01                BRW     169$
                           00000000G          EF  9F 00F04 44$:           PUSHAB  SHIFT                       ; 0771
                                              04  DD 00F0A                PUSHL   #4
                           00000000G          EF  9F 00F0C                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F12                CALLS   #3,PAS$WRITE_STRING
                           FFFFB8BB           EF  9F 00F19                PUSHAB  C.AES
                                              03  DD 00F1F                PUSHL   #3
                           00000000G          EF  9F 00F21                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F27                CALLS   #3,PAS$WRITE_STRING
                                              03  DD 00F2E                PUSHL   #3
                           00000084G          EF  DD 00F30                PUSHL   IDATA+132
                           00000000G          EF  9F 00F36                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F3C                CALLS   #3,PAS$WRITE_INTEGER
                           FFFFB895           EF  9F 00F43                PUSHAB  C.AET
                                              1D  DD 00F49                PUSHL   #29
                           00000000G          EF  9F 00F4B                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F51                CALLS   #3,PAS$WRITE_STRING
                           00000084G          EF  D5 00F58                TSTL    IDATA+132                   ; 0777
                                              00V 12  00F5E               BNEQ    46$
                           FFFFB898           EF  9F 00F60                PUSHAB  C.AEU                       ; 0779
                                              06  DD 00F66                PUSHL   #6
                           00000000G          EF  9F 00F68                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F6E                CALLS   #3,PAS$WRITE_STRING
                                         0000V  31  00F75                BRW     169$
                           FFFFB888           EF  9F 00F78 46$:           PUSHAB  C.AEV                       ; 0783
                                              07  DD 00F7E                PUSHL   #7
                           00000000G          EF  9F 00F80                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F86                CALLS   #3,PAS$WRITE_STRING
                                         0000V  31  00F8D                BRW     169$
                           00000000G          EF  9F 00F90 48$:           PUSHAB  SHIFT                       ; 0789
                                              04  DD 00F96                PUSHL   #4
                           00000000G          EF  9F 00F98                PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                 03  FB 00F9E                CALLS   #3,PAS$WRITE_STRING
                           FFFFB863           EF  9F 00FA5                PUSHAB  C.AEW
                                              03  DD 00FAB                PUSHL   #3
```

```
                 00000000G  EF   9F  00FAD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  00FB3          CALLS   #3,PAS$WRITE_STRING
                            03   DD  00FBA          PUSHL   #3
                 00000084G  EF   DD  00FBC          PUSHL   IDATA+132
                 00000000G  EF   9F  00FC2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  00FC8          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFB83D   EF   9F  00FCF          PUSHAB  C.AEX
                            25   DD  00FD5          PUSHL   #37
                 00000000G  EF   9F  00FD7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  00FDD          CALLS   #3,PAS$WRITE_STRING
                       0000V 31   00FE4          BRW     169$
                 00000000G  EF   9F  00FE7 49$:     PUSHAB  SHIFT                           ; 0795
                            04   DD  00FED          PUSHL   #4
                 00000000G  EF   9F  00FEF          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  00FF5          CALLS   #3,PAS$WRITE_STRING
                 FFFFB838   EF   9F  00FFC          PUSHAB  C.AEY
                            27   DD  01002          PUSHL   #39
                 00000000G  EF   9F  01004          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  0100A          CALLS   #3,PAS$WRITE_STRING
                       0000V 31   01011          BRW     169$
                 00000000G  EF   9F  01014 50$:     PUSHAB  SHIFT                           ; 0800
                            04   DD  0101A          PUSHL   #4
                 00000000G  EF   9F  0101C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  01022          CALLS   #3,PAS$WRITE_STRING
                 FFFFB833   EF   9F  01029          PUSHAB  C.AEZ
                            22   DD  0102F          PUSHL   #34
                 00000000G  EF   9F  01031          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  01037          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F  0103E          PUSHAB  ANSI_REVERSE
                            04   DD  01044          PUSHL   #4
                 00000000G  EF   9F  01046          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  0104C          CALLS   #3,PAS$WRITE_STRING
                 FFFFB82D   EF   9F  01053          PUSHAB  C.AFA
                            03   DD  01059          PUSHL   #3
                 00000000G  EF   9F  0105B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  01061          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F  01068          PUSHAB  ANSI_RESET
                            04   DD  0106E          PUSHL   #4
                 00000000G  EF   9F  01070          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  01076          CALLS   #3,PAS$WRITE_STRING
                 FFFFB807   EF   9F  0107D          PUSHAB  C.AFB
                            03   DD  01083          PUSHL   #3
                 00000000G  EF   9F  01085          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  0108B          CALLS   #3,PAS$WRITE_STRING
                       0000V 31   01092          BRW     169$
00V00000000G  EF            00   E0  01095 51$:     BBS     #0,OPTIMIZING,53$               ; 0807
                 00000000G  EF   9F  0109D          PUSHAB  SHIFT                           ; 0809
                            04   DD  010A3          PUSHL   #4
                 00000000G  EF   9F  010A5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  010AB          CALLS   #3,PAS$WRITE_STRING
                 FFFFB7D6   EF   9F  010B2          PUSHAB  C.AFC
                            2F   DD  010B8          PUSHL   #47
                 00000000G  EF   9F  010BA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB  010C0          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F  010C7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               01   FB  010CD          CALLS   #1,PAS$WRITELN2
                            00V  11  010D4          BRB     54$
```

```
                                Generated Code

                                00000000G EF 9F 010D6 53$:    PUSHAB  SHIFT                                   ; 0814
                                          04 DD 010DC         PUSHL   #4
                                00000000G EF 9F 010DE         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 010E4         CALLS   #3,PAS$WRITE_STRING
                                FFFFB7CD   EF 9F 010EB         PUSHAB  C.AFD
                                          27 DD 010F1         PUSHL   #39
                                00000000G EF 9F 010F3         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 010F9         CALLS   #3,PAS$WRITE_STRING
                                00000000G EF 9F 01100         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         01 FB 01106         CALLS   #1,PAS$WRITELN2
                                00000000G EF 9F 0110D 54$:    PUSHAB  SHIFT                                   ; 0816
                                          04 DD 01113         PUSHL   #4
                                00000000G EF 9F 01115         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 0111B         CALLS   #3,PAS$WRITE_STRING
                                FFFFB7BE   EF 9F 01122         PUSHAB  C.AFE
                                          0D DD 01128         PUSHL   #13
                                00000000G EF 9F 0112A         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 01130         CALLS   #3,PAS$WRITE_STRING
                                FFFFB7B9   EF 9F 01137         PUSHAB  C.AFF
                                          0C DD 0113D         PUSHL   #12
                                00000000G EF 9F 0113F         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 01145         CALLS   #3,PAS$WRITE_STRING
                  00V00000000G  EF         00 E0 0114C         BBS     #0,OPTIMIZING,56$                       ; 0818
                                00000000G EF 9F 01154         PUSHAB  ANSI_REVERSE                            ; 0820
                                          04 DD 0115A         PUSHL   #4
                                00000000G EF 9F 0115C         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 01162         CALLS   #3,PAS$WRITE_STRING
                                FFFFB793   EF 9F 01169         PUSHAB  C.AFG
                                          03 DD 0116F         PUSHL   #3
                                00000000G EF 9F 01171         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 01177         CALLS   #3,PAS$WRITE_STRING
                                00000000G EF 9F 0117E         PUSHAB  ANSI_RESET
                                          04 DD 01184         PUSHL   #4
                                00000000G EF 9F 01186         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 0118C         CALLS   #3,PAS$WRITE_STRING
                                FFFFB76D   EF 9F 01193         PUSHAB  C.AFH
                                          03 DD 01199         PUSHL   #3
                                00000000G EF 9F 0119B         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 011A1         CALLS   #3,PAS$WRITE_STRING
                                     0000V 31 011A8         BRW     169$
                                          01 DD 011AB 56$:    PUSHL   #1                                      ; 0826
                                7E      5B 8F 9A 011AD         MOVZBL  #91,-(SP)
                                00000000G EF 9F 011B1         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 011B7         CALLS   #3,PAS$WRITE_CHAR
                                00000000G EF 9F 011BE         PUSHAB  OLD_COUNT
                     00000000G  EF         01 FB 011C4         CALLS   #1,NUM_LEN
                                          50 DD 011CB         PUSHL   R0
                                00000000G EF DD 011CD         PUSHL   OLD_COUNT
                                00000000G EF 9F 011D3         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 011D9         CALLS   #3,PAS$WRITE_INTEGER
                                          01 DD 011E0         PUSHL   #1
                                7E      5D 8F 9A 011E2         MOVZBL  #93,-(SP)
                                00000000G EF 9F 011E6         PUSHAB  PAS$FV_OUTPUT
                     00000000G  EF         03 FB 011EC         CALLS   #3,PAS$WRITE_CHAR
                                00000000G EF 9F 011F3         PUSHAB  OLD_COUNT                               ; 0828
                     00000000G  EF         01 FB 011F9         CALLS   #1,NUM_LEN
                                04         50 D1 01200         CMPL    R0,#4
```

```
                          00V  15 01203          BLEQ    58$
                FFFFB6FF  EF   9F 01205          PUSHAB  C.AFI                                   ; 0830
                          03   DD 0120B          PUSHL   #3
                00000000G EF   9F 0120D          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 01213      CALLS   #3,PAS$WRITE_STRING
                         0000V  31 01216          BRW     169$
                FFFFB6EB  EF   9F 01219 58$:     PUSHAB  C.AFJ                                   ; 0834
                          03   DD 0121F          PUSHL   #3
                00000000G EF   9F 01221          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 01227      CALLS   #3,PAS$WRITE_STRING
                         0000V  31 0122E          BRW     169$
                00000000G EF   9F 01231 61$:     PUSHAB  SHIFT                                   ; 0844
                          04   DD 01237          PUSHL   #4
                00000000G EF   9F 01239          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 0123F      CALLS   #3,PAS$WRITE_STRING
                FFFFB6C2  EF   9F 01246          PUSHAB  C.AFK
                          26   DD 0124C          PUSHL   #38
                00000000G EF   9F 0124E          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 01254      CALLS   #3,PAS$WRITE_STRING
                00000000G EF   9F 0125B          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              01 FB 01261      CALLS   #1,PAS$WRITELN2
00V000000000G   EF              00 E0 01268      BBS     #0,OPTIMIZING,63$                       ; 0846
                00000000G EF   9F 01270          PUSHAB  SHIFT                                   ; 0848
                          04   DD 01276          PUSHL   #4
                00000000G EF   9F 01278          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 0127E      CALLS   #3,PAS$WRITE_STRING
                FFFFB6AB  EF   9F 01285          PUSHAB  C.AFL
                          19   DD 0128B          PUSHL   #25
                00000000G EF   9F 0128D          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 01293      CALLS   #3,PAS$WRITE_STRING
                          00V  11 0129A          BRB     64$
                00000000G EF   9F 0129C 63$:     PUSHAB  SHIFT                                   ; 0852
                          04   DD 012A2          PUSHL   #4
                00000000G EF   9F 012A4          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 012AA      CALLS   #3,PAS$WRITE_STRING
                FFFFB69B  EF   9F 012B1          PUSHAB  C.AFM
                          17   DD 012B7          PUSHL   #23
                00000000G EF   9F 012B9          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 012BF      CALLS   #3,PAS$WRITE_STRING
                FFFFB69E  EF   9F 012C6 64$:     PUSHAB  C.AFN                                   ; 0854
                          12   DD 012CC          PUSHL   #18
                00000000G EF   9F 012CE          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 012D4      CALLS   #3,PAS$WRITE_STRING
                         0000V  31 012DB          BRW     169$
00V000000000G   EF              00 E0 012DE 65$: BBS     #0,OPTIMIZING,67$                       ; 0862
                00000000G EF   9F 012E6          PUSHAB  SHIFT                                   ; 0864
                          04   DD 012EC          PUSHL   #4
                00000000G EF   9F 012EE          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 012F4      CALLS   #3,PAS$WRITE_STRING
                FFFFB67D  EF   9F 012FB          PUSHAB  C.AFO
                          31   DD 01301          PUSHL   #49
                00000000G EF   9F 01303          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03 FB 01309      CALLS   #3,PAS$WRITE_STRING
                00000000G EF   9F 01310          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              01 FB 01316      CALLS   #1,PAS$WRITELN2
                          00V  11 0131D          BRB     68$
                00000000G EF   9F 0131F 67$:     PUSHAB  SHIFT                                   ; 0869
```

Generated Code

```
                                04   DD 01329          PUSHL   #4
                    00000000G   EF   9F 0132B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01331          CALLS   #3,PAS$WRITE_STRING
                    FFFFB678    EF   9F 01338          PUSHAB  C.AFP
                                2F   DD 0133E          PUSHL   #47
                    00000000G   EF   9F 01340          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01346          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 0134D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01   FB 01353          CALLS   #1,PAS$WRITELN2
                    00000000G   EF   9F 0135A 68$:      PUSHAB  SHIFT                           ; 0872
                                04   DD 01360          PUSHL   #4
                    00000000G   EF   9F 01362          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01368          CALLS   #3,PAS$WRITE_STRING
                    FFFFB671    EF   9F 0136F          PUSHAB  C.AFQ
                                28   DD 01375          PUSHL   #40
                    00000000G   EF   9F 01377          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0137D          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   01384          BRW     169$
                    00000000G   EF   9F 01387 69$:      PUSHAB  SHIFT                           ; 0881
                                04   DD 0138D          PUSHL   #4
                    00000000G   EF   9F 0138F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01395          CALLS   #3,PAS$WRITE_STRING
                    FFFFB66C    EF   9F 0139C          PUSHAB  C.AFR
                                2E   DD 013A2          PUSHL   #46
                    00000000G   EF   9F 013A4          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 013AA          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 013B1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01   FB 013B7          CALLS   #1,PAS$WRITELN2
00V000000000G   EF              00   E0 013BE          BBS     #0,OPTIMIZING,71$               ; 0884
                    00000000G   EF   9F 013C6          PUSHAB  SHIFT                           ; 0886
                                04   DD 013CC          PUSHL   #4
                    00000000G   EF   9F 013CE          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 013D4          CALLS   #3,PAS$WRITE_STRING
                    FFFFB65D    EF   9F 013DB          PUSHAB  C.AFS
                                15   DD 013E1          PUSHL   #21
                    00000000G   EF   9F 013E3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 013E9          CALLS   #3,PAS$WRITE_STRING
                                00V  11 013F0          BRB     72$
                    00000000G   EF   9F 013F2 71$:      PUSHAB  SHIFT                           ; 0890
                                04   DD 013F8          PUSHL   #4
                    00000000G   EF   9F 013FA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01400          CALLS   #3,PAS$WRITE_STRING
                    FFFFB649    EF   9F 01407          PUSHAB  C.AFT
                                16   DD 0140D          PUSHL   #22
                    00000000G   EF   9F 0140F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01415          CALLS   #3,PAS$WRITE_STRING
                    FFFFB64C    EF   9F 0141C 72$:      PUSHAB  C.AFU                           ; 0892
                                11   DD 01422          PUSHL   #17
                    00000000G   EF   9F 01424          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0142A          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   01431          BRW     169$
                    00000000G   EF   9F 01434 73$:      PUSHAB  SHIFT                           ; 0898
                                04   DD 0143A          PUSHL   #4
                    00000000G   EF   9F 0143C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01442          CALLS   #3,PAS$WRITE_STRING
                    FFFFB633    EF   9F 01449          PUSHAB  C.AFV
                                2D   DD 0144F          PUSHL   #45
```

```
                    00000000G  EF   9F 01451          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01457          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 0145E          BRW     169$
                    00000000G  EF   9F 01461 74$:     PUSHAB  SHIFT                                   ; 0903
                               04   DD 01467          PUSHL   #4
                    00000000G  EF   9F 01469          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 0146F          CALLS   #3,PAS$WRITE_STRING
                    FFFFB636   EF   9F 01476          PUSHAB  C.AFW
                               30   DD 0147C          PUSHL   #48
                    00000000G  EF   9F 0147E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01484          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 0148B          BRW     169$
                    00000000G  EF   9F 0148E 75$:     PUSHAB  SHIFT                                   ; 0908
                               04   DD 01494          PUSHL   #4
                    00000000G  EF   9F 01496          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 0149C          CALLS   #3,PAS$WRITE_STRING
                    FFFFB639   EF   9F 014A3          PUSHAB  C.AFX
                               2A   DD 014A9          PUSHL   #42
                    00000000G  EF   9F 014AB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 014B1          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 014B8          BRW     169$
                    00000000G  EF   9F 014BB 76$:     PUSHAB  SHIFT                                   ; 0913
                               04   DD 014C1          PUSHL   #4
                    00000000G  EF   9F 014C3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 014C9          CALLS   #3,PAS$WRITE_STRING
                    FFFFB638   EF   9F 014D0          PUSHAB  C.AFY
                               2F   DD 014D6          PUSHL   #47
                    00000000G  EF   9F 014D8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 014DE          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 014E5          BRW     169$
                    00000000G  EF   9F 014E8 77$:     PUSHAB  SHIFT                                   ; 0918
                               04   DD 014EE          PUSHL   #4
                    00000000G  EF   9F 014F0          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 014F6          CALLS   #3,PAS$WRITE_STRING
                    FFFFB63B   EF   9F 014FD          PUSHAB  C.AFZ
                               2D   DD 01503          PUSHL   #45
                    00000000G  EF   9F 01505          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 0150B          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 01512          BRW     169$
                    00000000G  EF   9F 01515 78$:     PUSHAB  SHIFT                                   ; 0923
                               04   DD 0151B          PUSHL   #4
                    00000000G  EF   9F 0151D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01523          CALLS   #3,PAS$WRITE_STRING
                    FFFFB63E   EF   9F 0152A          PUSHAB  C.AGA
                               2D   DD 01530          PUSHL   #45
                    00000000G  EF   9F 01532          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01538          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0153F          PUSHAB  CRLF_SHIFT
                               06   DD 01545          PUSHL   #6
                    00000000G  EF   9F 01547          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 0154D          CALLS   #3,PAS$WRITE_STRING
                    FFFFB644   EF   9F 01554          PUSHAB  C.AGB
                               2E   DD 0155A          PUSHL   #46
                    00000000G  EF   9F 0155C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01562          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 01569          BRW     169$
                    00000000G  EF   9F 0156C 79$:     PUSHAB  SHIFT                                   ; 0929
```

```
                                 04    DD 01572          PUSHL   #4
                    00000000G    EF    9F 01574          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 0157A          CALLS   #3,PAS$WRITE_STRING
                    FFFFB647     EF    9F 01581          PUSHAB  C.AGC
                                 20    DD 01587          PUSHL   #32
                    00000000G    EF    9F 01589          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 0158F          CALLS   #3,PAS$WRITE_STRING
                           0000V 31    01596          BRW     169$
                    00000000G    EF    9F 01599 80$:     PUSHAB  SHIFT                                   ; 0936
                                 04    DD 0159F          PUSHL   #4
                    00000000G    EF    9F 015A1          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 015A7          CALLS   #3,PAS$WRITE_STRING
                    FFFFB63A     EF    9F 015AE          PUSHAB  C.AGD
                                 03    DD 015B4          PUSHL   #3
                    00000000G    EF    9F 015B6          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 015BC          CALLS   #3,PAS$WRITE_STRING
                                 03    DD 015C3          PUSHL   #3
                    00000084G    EF    DD 015C5          PUSHL   IDATA+132
                    00000000G    EF    9F 015CB          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 015D1          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB614     EF    9F 015D8          PUSHAB  C.AGE
                                 09    DD 015DE          PUSHL   #9
                    00000000G    EF    9F 015E0          PUSHAB  PAS$FV_OUTPUT
             00000000G  EF       03    FB 015E6          CALLS   #3,PAS$WRITE_STRING
00V00000013G  EF                 00    E1 015ED          BBC     #0,BDATA+19,82$                         ; 0939
                    FFFFB603     EF    9F 015F5          PUSHAB  C.AGF                                   ; 0941
                                 03    DD 015FB          PUSHL   #3
                    00000000G    EF    9F 015FD          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 01603          CALLS   #3,PAS$WRITE_STRING
                                 01    DD 0160A          PUSHL   #1
                    00000000G    EF    DD 0160C          PUSHL   SEGMENT_NUMBER
                    00000000G    EF    9F 01612          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 01618          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB5DD     EF    9F 0161F 82$:     PUSHAB  C.AGG                                   ; 0943
                                 02    DD 01625          PUSHL   #2
                    00000000G    EF    9F 01627          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 0162D          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF    9F 01634          PUSHAB  MIN_KEY_SIZE
                00000000G  EF    01    FB 0163A          CALLS   #1,NUM_CEN
                                 50    DD 01641          PUSHL   R0
                    00000000G    EF    DD 01643          PUSHL   MIN_KEY_SIZE
                    00000000G    EF    9F 01649          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 0164F          CALLS   #3,PAS$WRITE_INTEGER
                                 01    DD 01656          PUSHL   #1
                                 2D    DD 01658          PUSHL   #45
                    00000000G    EF    9F 0165A          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 01660          CALLS   #3,PAS$WRITE_CHAR
                    00000000G    EF    9F 01667          PUSHAB  MAX_KEY_SIZE
                00000000G  EF    01    FB 0166D          CALLS   #1,NUM_CEN
                                 50    DD 01674          PUSHL   R0
                    0000000UG    EF    DD 01676          PUSHL   MAX_KEY_SIZE
                    00000000G    EF    9F 0167C          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 01682          CALLS   #3,PAS$WRITE_INTEGER
                                 01    DD 01689          PUSHL   #1
                                 29    DD 0168B          PUSHL   #41
                    00000000G    EF    9F 0168D          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03    FB 01693          CALLS   #3,PAS$WRITE_CHAR
```

```
                    00000000G  EF   9F 0169A          PUSHAB  ANSI_REVERSE
                               04   DD 016A0          PUSHL   #4
                    00000000G  EF   9F 016A2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 016A8          CALLS   #3,PAS$WRITE_STRING
                    FFFFB551   EF   9F 016AF          PUSHAB  C.AGH
                               03   DD 016B5          PUSHL   #3
                    00000000G  EF   9F 016B7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 016BD          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 016C4          PUSHAB  ANSI_RESET
                               04   DD 016CA          PUSHL   #4
                    00000000G  EF   9F 016CC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 016D2          CALLS   #3,PAS$WRITE_STRING
                    FFFFB52B   EF   9F 016D9          PUSHAB  C.AGI
                               03   DD 016DF          PUSHL   #3
                    00000000G  EF   9F 016E1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 016E7          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   16EE             BRW     169$
                    00000000G  EF   9F 016F1 83$:     PUSHAB  SHIFT                          ; 0953
                               04   DD 016F7          PUSHL   #4
                    00000000G  EF   9F 016F9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 016FF          CALLS   #3,PAS$WRITE_STRING
                    FFFFB502   EF   9F 01706          PUSHAB  C.AGJ
                               25   DD 0170C          PUSHL   #37
                    00000000G  EF   9F 0170E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01714          CALLS   #3,PAS$WRITE_STRING
                    00000098G  EF   D5 0171B          TSTL    IDATA+152                      ; 0955
                               00V  12 01721          BNEQ    85$
                    FFFFB50D   EF   9F 01723          PUSHAB  C.AGK                          ; 0957
                               14   DD 01729          PUSHL   #20
                    00000000G  EF   9F 0172B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01731          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 01738          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  01   FB 0173E          CALLS   #1,PAS$WRITELN2
                               00V  11 01745          BRB     86$
                    FFFFB4FD   EF   9F 01747 85$:     PUSHAB  C.AGL                          ; 0961
                               14   DD 0174D          PUSHL   #20
                    00000000G  EF   9F 0174F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01755          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0175C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  01   FB 01762          CALLS   #1,PAS$WRITELN2
                    00000000G  EF   9F 01769 86$:     PUSHAB  SHIFT                          ; 0963
                               04   DD 0176F          PUSHL   #4
                    00000000G  EF   9F 01771          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 01777          CALLS   #3,PAS$WRITE_STRING
                    FFFFB4DA   EF   9F 0177E          PUSHAB  C.AGM
                               1C   DD 01784          PUSHL   #28
                    00000000G  EF   9F 01786          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 0178C          CALLS   #3,PAS$WRITE_STRING
                               06   DD 01793          PUSHL   #6
                    00000000G  EF   DD 01795          PUSHL   BREAKPOINT_LEFT
                    00000000G  EF   9F 0179B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 017A1          CALLS   #3,PAS$WRITE_INTEGER
                               06   DD 017A8          PUSHL   #6
                    00000000G  EF   DD 017AA          PUSHL   BREAKPOINT_MID
                    00000000G  EF   9F 017B0          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03   FB 017B6          CALLS   #3,PAS$WRITE_INTEGER
                               06   DD 017BD          PUSHL   #6
```

```
                    00000000G  EF    DD 017BF          PUSHL   BREAKPOINT_RIGHT
                    00000000G  EF    9F 017C5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 017CB          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB4A2   EF    9F 017D2          PUSHAB  C.AGN
                               02    DD 017D8          PUSHL   #2
                    00000000G  EF    9F 017DA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 017E0          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 017E7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 01    FB 017ED          CALLS   #1,PAS$WRITELN2
                    00000000G  EF    9F 017F4          PUSHAB  SHIFT                                   ; 0967
                               04    DD 017FA          PUSHL   #4
                    00000000G  EF    9F 017FC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 01802          CALLS   #3,PAS$WRITE_STRING
                    FFFFB46F   EF    9F 01809          PUSHAB  C.AGO
                               1E    DD 0180F          PUSHL   #30
                    00000000G  EF    9F 01811          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 01817          CALLS   #3,PAS$WRITE_STRING
                               06    DD 0181E          PUSHL   #6
                    00000000G  EF    DD 01820          PUSHL   DEPTHPOINT_LEFT
                    00000000G  EF    9F 01826          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 0182C          CALLS   #3,PAS$WRITE_INTEGER
                               06    DD 01833          PUSHL   #6
                    00000000G  EF    DD 01835          PUSHL   DEPTHPOINT_MID
                    00000000G  EF    9F 0183B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 01841          CALLS   #3,PAS$WRITE_INTEGER
                               06    DD 01848          PUSHL   #6
                    00000000G  EF    DD 0184A          PUSHL   DEPTHPOINT_RIGHT
                    00000000G  EF    9F 01850          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 01856          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB439   EF    9F 0185D          PUSHAB  C.AGP
                               02    DD 01863          PUSHL   #2
                    00000000G  EF    9F 01865          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 0186B          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 01872          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 01    FB 01878          CALLS   #1,PAS$WRITELN2
                    00000000G  EF    9F 0187F          PUSHAB  SHIFT                                   ; 0971
                               04    DD 01885          PUSHL   #4
                    00000000G  EF    9F 01887          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 0188D          CALLS   #3,PAS$WRITE_STRING
                    FFFFB404   EF    9F 01894          PUSHAB  C.AGQ
                               1F    DD 0189A          PUSHL   #31
                    00000000G  EF    9F 0189C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 018A2          CALLS   #3,PAS$WRITE_STRING
                               06    DD 018A9          PUSHL   #6
                    00000C00G  EF    DD 018AB          PUSHL   NUMPOINT_LEFT
                    00000000G  EF    9F 018B1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 018B7          CALLS   #3,PAS$WRITE_INTEGER
                               06    DD 018BE          PUSHL   #6
                    00000000G  EF    DD 018C0          PUSHL   NUMPOINT_MID
                    00000000G  EF    9F 018C6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 018CC          CALLS   #3,PAS$WRITE_INTEGER
                               06    DD 018D3          PUSHL   #6
                    00000000G  EF    DD 018D5          PUSHL   NUMPOINT_RIGHT
                    00000000G  EF    9F 018DB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                 03    FB 018E1          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB3D0   EF    9F 018E8          PUSHAB  C.AGR
                               02    DD 018EE          PUSHL   #2
```

Generated Code

```
                 00000000G  EF  9F  018F0          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  018F6          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F  018FD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                01  FB  01903          CALLS   #1,PAS$WRITELN2
                 00000000G   EF  9F  0190A          PUSHAB  SHIFT                          ; 0975
                             04  DD  01910          PUSHL   #4
                 00000000G   EF  9F  01912          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  01918          CALLS   #3,PAS$WRITE_STRING
                 FFFFB39D    EF  9F  0191F          PUSHAB  C_AGS
                             22  DD  01925          PUSHL   #34
                 00000000G   EF  9F  01927          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  0192D          CALLS   #3,PAS$WRITE_STRING
                             06  DD  01934          PUSHL   #6
                 00000000G   EF  DD  01936          PUSHL   PAGEPOINT_LEFT
                 00000000G   EF  9F  0193C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  01942          CALLS   #3,PAS$WRITE_INTEGER
                             06  DD  01949          PUSHL   #6
                 00000000G   EF  DD  0194B          PUSHL   PAGEPOINT_MID
                 00000000G   EF  9F  01951          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  01957          CALLS   #3,PAS$WRITE_INTEGER
                             06  DD  0195E          PUSHL   #6
                 00000000G   EF  DD  01960          PUSHL   PAGEPOINT_RIGHT
                 00000000G   EF  9F  01966          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  0196C          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFB368    EF  9F  01973          PUSHAB  C_AGT
                             02  DD  01979          PUSHL   #2
                 00000000G   EF  9F  0197B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  01981          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F  01988          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                01  FB  0198E          CALLS   #1,PAS$WRITELN2
                 00000000G   EF  9F  01995          PUSHAB  SHIFT                          ; 0979
                             04  DD  0199B          PUSHL   #4
                 00000000G   EF  9F  0199D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  019A3          CALLS   #3,PAS$WRITE_STRING
                 FFFFB336    EF  9F  019AA          PUSHAB  C_AGU
                             23  DD  019B0          PUSHL   #35
                 00000000G   EF  9F  019B2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  019B8          CALLS   #3,PAS$WRITE_STRING
                             06  DD  019BF          PUSHL   #6
                 00000000G   EF  DD  019C1          PUSHL   EXAMPOINT_LEFT
                 00000000G   EF  9F  019C7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  019CD          CALLS   #3,PAS$WRITE_INTEGER
                             06  DD  019D4          PUSHL   #6
                 00000000G   EF  DD  019D6          PUSHL   EXAMPOINT_MID
                 00000000G   EF  9F  019DC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  019E2          CALLS   #3,PAS$WRITE_INTEGER
                             06  DD  019E9          PUSHL   #6
                 00000000G   EF  DD  019EB          PUSHL   EXAMPOINT_RIGHT
                 00000000G   EF  9F  019F1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  019F7          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFB306    EF  9F  019FE          PUSHAB  C_AGV
                             02  DD  01A04          PUSHL   #2
                 00000000G   EF  9F  01A06          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB  01A0C          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F  01A13          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                01  FB  01A19          CALLS   #1,PAS$WRITELN2
                 00000000G   EF  9F  01A20          PUSHAB  CRLF_SHIFT                     ; 0983
```

```
                                                 06   DD 01A26          PUSHL   #6
                                 00000000G       EF   9F 01A28          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01A2E          CALLS   #3,PAS$WRITE_STRING
                                  FFFFB2D3       EF   9F 01A35          PUSHAB  C.AGW
                                                 03   DD 01A3B          PUSHL   #3
                                 00000000G       EF   9F 01A3D          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01A43          CALLS   #3,PAS$WRITE_STRING
                                                 03   DD 01A4A          PUSHL   #3
                                  00000084G      EF   DD 01A4C          PUSHL   IDATA+132
                                 00000000G       EF   9F 01A52          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01A58          CALLS   #3,PAS$WRITE_INTEGER
                                  FFFFB2AD       EF   9F 01A5F          PUSHAB  C.AGX
                                                 0F   DD 01A65          PUSHL   #15
                                 00000000G       EF   9F 01A67          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01A6D          CALLS   #3,PAS$WRITE_STRING
                                 00000000G       EF   9F 01A74          PUSHAB  MIN_BUCKET
                        00000000G EF             01   FB 01A7A          CALLS   #1,NUM_LEN
                                                 50   DD 01A81          PUSHL   R0
                                  00000000G      EF   DD 01A83          PUSHL   MIN_BUCKET
                                 00000000G       EF   9F 01A89          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01A8F          CALLS   #3,PAS$WRITE_INTEGER
                                  FFFFB286       EF   9F 01A96          PUSHAB  C.AGY
                                                 05   DD 01A9C          PUSHL   #5
                                 00000000G       EF   9F 01A9E          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01AA4          CALLS   #3,PAS$WRITE_STRING
                     50 00000000G EF             19   C5 01AAB          MULL3   #25,QTAB_OFFSET,R0
        5C FFFFFEF2GEF40          20             00   EE 01AB3          EXTV    #0,#32,QTAB-270[R0],R12
                               FC AD             5C   D0 01ABD          MOVL    R12,-4(FP)
                                     FC AD       9F   9A 01AC1          PUSHAB  -4(FP)
                        00000000G EF             01   FB 01AC4          CALLS   #1,NUM_LEN
                                                 50   DD 01ACB          PUSHL   R0
                                                 5C   DD 01ACD          PUSHL   R12
                                 00000000G       EF   9F 01ACF          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01AD5          CALLS   #3,PAS$WRITE_INTEGER
                                  FFFFB248       EF   9F 01ADC          PUSHAB  C.AGZ
                                                 04   DD 01AE2          PUSHL   #4
                                 00000000G       EF   9F 01AE4          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01AEA          CALLS   #3,PAS$WRITE_STRING
                                           0000V 31   01AF1          BRW     169$
                                  00000000G      EF   9F 01AF4 87$:     PUSHAB  SHIFT                                   ; 0996
                                                 04   DD 01AFA          PUSHL   #4
                                 00000000G       EF   9F 01AFC          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01B02          CALLS   #3,PAS$WRITE_STRING
                                  FFFFB21F       EF   9F 01B09          PUSHAB  C.AHA
                                                 1F   DD 01B0F          PUSHL   #31
                                 00000000G       EF   9F 01B11          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01B17          CALLS   #3,PAS$WRITE_STRING
                                 00000000G       EF   9F 01B1E          PUSHAB  CRLF_SHIFT
                                                 06   DD 01B24          PUSHL   #6
                                 00000000G       EF   9F 01B26          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01B2C          CALLS   #3,PAS$WRITE_STRING
                                  FFFFB215       EF   9F 01B33          PUSHAB  C.AHB
                                                 2A   DD 01B39          PUSHL   #42
                                 00000000G       EF   9F 01B3B          PUSHAB  PAS$FV_OUTPUT
                        00000000G EF             03   FB 01B41          CALLS   #3,PAS$WRITE_STRING
                     50 00000000G EF             19   C5 01B48          MULL3   #25,QTAB_OFFSET,R0                      ; 0999
        01 FFFFFEF2GEF40          20             00   EC 01B50          CMPV    #0,#32,QTAB-270[R0],#1
```

```
                    00V  12 01B5A          BNEQ    89$
            FFFFB218    EF  9F 01B5C          PUSHAB  C.AHC                          ; 1001
                        08  DD 01B62          PUSHL   #8
            00000000G   EF  9F 01B64          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01B6A      CALLS   #3,PAS$WRITE_STRING
                      0000V 31 01B71          BRW     169$
            FFFFB208    EF  9F 01B74 89$:     PUSHAB  C.AHD                          ; 1005
                        09  DD 01B7A          PUSHL   #9
            00000000G   EF  9F 01B7C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01B82      CALLS   #3,PAS$WRITE_STRING
                      0000V 31 01B89          BRW     169$
            00000000G   EF  9F 01B8C 91$:     PUSHAB  SHIFT                          ; 1011
                        04  DD 01B92          PUSHL   #4
            00000000G   EF  9F 01B94          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01B9A      CALLS   #3,PAS$WRITE_STRING
            FFFFB1E7    EF  9F 01BA1          PUSHAB  C.AHE
                        03  DD 01BA7          PUSHL   #3
            00000000G   EF  9F 01BA9          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01BAF      CALLS   #3,PAS$WRITE_STRING
                        03  DD 01BB6          PUSHL   #3
            00000084G   EF  DD 01BB8          PUSHL   IDATA+132
            00000000G   EF  9F 01BBE          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01BC4      CALLS   #3,PAS$WRITE_INTEGER
            FFFFB1C1    EF  9F 01BCB          PUSHAB  C.AHF
                        23  DD 01BD1          PUSHL   #35
            00000000G   EF  9F 01BD3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01BD9      CALLS   #3,PAS$WRITE_STRING
                      0000V 31 01BE0          BRW     169$
            00000000G   EF  9F 01BE3 92$:     PUSHAB  SHIFT                          ; 1016
                        04  DD 01BE9          PUSHL   #4
            00000000G   EF  9F 01BEB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01BF1      CALLS   #3,PAS$WRITE_STRING
            FFFFB1B8    EF  9F 01BF8          PUSHAB  C.AHG
                        2F  DD 01BFE          PUSHL   #47
            00000000G   EF  9F 01C00          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01C06      CALLS   #3,PAS$WRITE_STRING
                      0000V 31 01C0D          BRW     169$
            00000000G   EF  9F 01C10 93$:     PUSHAB  SHIFT                          ; 1021
                        04  DD 01C16          PUSHL   #4
            00000000G   EF  9F 01C18          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01C1E      CALLS   #3,PAS$WRITE_STRING
            FFFFB1B8    EF  9F 01C25          PUSHAB  C.AHH
                        28  DD 01C2B          PUSHL   #40
            00000000G   EF  9F 01C2D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01C33      CALLS   #3,PAS$WRITE_STRING
            00000000G   EF  9F 01C3A          PUSHAB  CRLF_SHIFT
                        06  DD 01C40          PUSHL   #6
            00000000G   EF  9F 01C42          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01C48      CALLS   #3,PAS$WRITE_STRING
            FFFFB1B9    EF  9F 01C4F          PUSHAB  C.AHI
                        02  DD 01C55          PUSHL   #2
            00000000G   EF  9F 01C57          PUSHAB  PAS$FV_OUTPUT
00000000G   EF              03  FB 01C5D      CALLS   #3,PAS$WRITE_STRING
                      0000V 31 01C64          BRW     169$
            00000000G   EF  9F 01C67 94$:     PUSHAB  SHIFT                          ; 1027
                        04  DD 01C6D          PUSHL   #4
            00000000G   EF  9F 01C6F          PUSHAB  PAS$FV_OUTPUT
```

```
                 00000000G  EF  03  FB 01C75          CALLS   #3,PAS$WRITE_STRING
                     FFFFB190   EF  9F 01C7C          PUSHAB  C.AHJ
                                2B  DD 01C82          PUSHL   #43
                     00000000G  EF  9F 01C84          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01C8A      CALLS   #3,PAS$WRITE_STRING
                     00000000G  EF  9F 01C91          PUSHAB  CRLF_SHIFT
                                06  DD 01C97          PUSHL   #6
                     00000000G  EF  9F 01C99          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01C9F      CALLS   #3,PAS$WRITE_STRING
                     FFFFB192   EF  9F 01CA6          PUSHAB  C.AHK
                                02  DD 01CAC          PUSHL   #2
                     00000000G  EF  9F 01CAE          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01CB4      CALLS   #3,PAS$WRITE_STRING
                               0000V 31 01CBB          BRW     169$
                     00000000G  EF  9F 01CBE 95$:     PUSHAB  SHIFT                   ; 1032
                                04  DD 01CC4          PUSHL   #4
                     00000000G  EF  9F 01CC6          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01CCC      CALLS   #3,PAS$WRITE_STRING
                     FFFFB169   EF  9F 01CD3          PUSHAB  C.AHL
                                2A  DD 01CD9          PUSHL   #42
                     00000000G  EF  9F 01CDB          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01CE1      CALLS   #3,PAS$WRITE_STRING
                     00000000G  EF  9F 01CE8          PUSHAB  CRLF_SHIFT
                                06  DD 01CEE          PUSHL   #6
                     00000000G  EF  9F 01CF0          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01CF6      CALLS   #3,PAS$WRITE_STRING
                     FFFFB169   EF  9F 01CFD          PUSHAB  C.AHM
                                02  DD 01D03          PUSHL   #2
                     00000000G  EF  9F 01D05          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D0B      CALLS   #3,PAS$WRITE_STRING
                               0000V 31 01D12          BRW     169$
                     00000000G  EF  9F 01D15 96$:     PUSHAB  SHIFT                   ; 1038
                                04  DD 01D1B          PUSHL   #4
                     00000000G  EF  9F 01D1D          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D23      CALLS   #3,PAS$WRITE_STRING
                     FFFFB13E   EF  9F 01D2A          PUSHAB  C.AHN
                                2E  DD 01D30          PUSHL   #46
                     00000000G  EF  9F 01D32          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D38      CALLS   #3,PAS$WRITE_STRING
                     00000000G  EF  9F 01D3F          PUSHAB  CRLF_SHIFT
                                06  DD 01D45          PUSHL   #6
                     00000000G  EF  9F 01D47          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D4D      CALLS   #3,PAS$WRITE_STRING
                     FFFFB142   EF  9F 01D54          PUSHAB  C.AHO
                                02  DD 01D5A          PUSHL   #2
                     00000000G  EF  9F 01D5C          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D62      CALLS   #3,PAS$WRITE_STRING
                               0000V 31 01D69          BRW     169$
                     00000000G  EF  9F 01D6C 97$:     PUSHAB  SHIFT                   ; 1044
                                04  DD 01D72          PUSHL   #4
                     00000000G  EF  9F 01D74          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D7A      CALLS   #3,PAS$WRITE_STRING
                     FFFFB117   EF  9F 01D81          PUSHAB  C.AHP
                                03  DD 01D87          PUSHL   #3
                     00000000G  EF  9F 01D89          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF      03  FB 01D8F      CALLS   #3,PAS$WRITE_STRING
                                03  DD 01D96          PUSHL   #3
```

```
                    00000084G  EF  DD 01D98          PUSHL   IDATA+132
                    00000000G  EF  9F 01D9E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01DA4          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB0F1   EF  9F 01DAB          PUSHAB  C.AHQ
                               1A  DD 01DB1          PUSHL   #26
                    00000000G  EF  9F 01DB3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01DB9          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 01DC0          PUSHAB  CRLF_SHIFT
                               06  DD 01DC6          PUSHL   #6
                    00000000G  EF  9F 01DC8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01DCE          CALLS   #3,PAS$WRITE_STRING
                    FFFFB0E1   EF  9F 01DD5          PUSHAB  C.AHR
                               02  DD 01DDB          PUSHL   #2
                    00000000G  EF  9F 01DDD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01DE3          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 01DEA          BRW     169$
                    00000000G  EF  9F 01DED 98$:     PUSHAB  SHIFT                                   ; 1052
                               04  DD 01DF3          PUSHL   #4
                    00000000G  EF  9F 01DF5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01DFB          CALLS   #3,PAS$WRITE_STRING
                    FFFFB0B6   EF  9F 01E02          PUSHAB  C.AHS
                               2E  DD 01E08          PUSHL   #46
                    00000000G  EF  9F 01E0A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E10          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 01E17          PUSHAB  CRLF_SHIFT
                               06  DD 01E1D          PUSHL   #6
                    00000000G  EF  9F 01E1F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E25          CALLS   #3,PAS$WRITE_STRING
                    FFFFB0BC   EF  9F 01E2C          PUSHAB  C.AHT
                               03  DD 01E32          PUSHL   #3
                    00000000G  EF  9F 01E34          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E3A          CALLS   #3,PAS$WRITE_STRING
                               03  DD 01E41          PUSHL   #3
                    00000084G  EF  DD 01E43          PUSHL   IDATA+132
                    00000000G  EF  9F 01E49          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E4F          CALLS   #3,PAS$WRITE_INTEGER
                    FFFFB096   EF  9F 01E56          PUSHAB  C.AHU
                               1D  DD 01E5C          PUSHL   #29
                    00000000G  EF  9F 01E5E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E64          CALLS   #3,PAS$WRITE_STRING
                          0000V    31 01E6B          BRW     169$
                    00000000G  EF  9F 01E6E 99$:     PUSHAB  SHIFT                                   ; 1060
                               04  DD 01E74          PUSHL   #4
                    00000000G  EF  9F 01E76          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E7C          CALLS   #3,PAS$WRITE_STRING
                    FFFFB089   EF  9F 01E83          PUSHAB  C.AHV
                               13  DD 01E89          PUSHL   #19
                    00000000G  EF  9F 01E8B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01E91          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 01E98          PUSHAB  LOW_KEY
00000000G  EF                  01  FB 01E9E          CALLS   #1,NUM_LEN
                               50  DD 01EA5          PUSHL   R0
                    00000000G  EF  DD 01EA7          PUSHL   LOW_KEY
                    00000000G  EF  9F 01EAD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                  03  FB 01EB3          CALLS   #3,PAS$WRITE_INTEGER
                               01  DD 01EBA          PUSHL   #1
                               2D  DD 01EBC          PUSHL   #45
```

```
                                00000000G  EF   9F  01EBE           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01EC4           CALLS   #3,PAS$WRITE_CHAR
                                00000000G  EF   9F  01ECB           PUSHAB  HIGH_KEY
                    00000000G   EF           01  FB  01ED1           CALLS   #1,NUM_LEN
                                             50  DD  01ED8           PUSHL   R0
                                00000000G  EF   DD  01EDA           PUSHL   HIGH_KEY
                                00000000G  EF   9F  01EE0           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01EE6           CALLS   #3,PAS$WRITE_INTEGER
                                FFFFB033   EF   9F  01EED           PUSHAB  C.AHW
                                           07   DD  01EF3           PUSHL   #7
                                00000000G  EF   9F  01EF5           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01EFB           CALLS   #3,PAS$WRITE_STRING
                                      0000V  31   01F02           BRW     169$
                                00000000G  EF   9F  01F05 100$:     PUSHAB  SHIFT                                   ; 1068
                                           04   DD  01F0B           PUSHL   #4
                                00000000G  EF   9F  01F0D           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01F13           CALLS   #3,PAS$WRITE_STRING
                                FFFFB00E   EF   9F  01F1A           PUSHAB  C.AHX
                                           21   DD  01F20           PUSHL   #33
                                00000000G  EF   9F  01F22           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01F28           CALLS   #3,PAS$WRITE_STRING
                 50 00000000G   EF           19  C5  01F2F           MULL3   #25,QTAB_OFFSET,R0
     5C FFFFFEF2GEF40           20           00  EE  01F37           EXTV    #0,#32,QTAB-270[R0],R12
                       FC   AD                  5C  D0  01F41           MOVL    R12,-4(FP)
                                FC   AD         9F  01F45           PUSHAB  -4(FP)
                    00000000G   EF           01  FB  01F48           CALLS   #1,NUM_LEN
                                             50  DD  01F4F           PUSHL   R0
                                             5C  DD  01F51           PUSHL   R12
                                00000000G  EF   9F  01F53           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           C3  FB  01F59           CALLS   #3,PAS$WRITE_INTEGER
                                FFFFAFEC   EF   9F  01F60           PUSHAB  C.AHY
                                           04   DD  01F66           PUSHL   #4
                                00000000G  EF   9F  01F68           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01F6E           CALLS   #3,PAS$WRITE_STRING
                                      0000V  31   01F75           BRW     169$
                                00000000G  EF   9F  01F78 101$:     PUSHAB  SHIFT                                   ; 1078
                                           04   DD  01F7E           PUSHL   #4
                                00000000G  EF   9F  01F80           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01F86           CALLS   #3,PAS$WRITE_STRING
                                FFFFAFC3   EF   9F  01F8D           PUSHAB  C.AHZ
                                           24   DD  01F93           PUSHL   #36
                                00000000G  EF   9F  01F95           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01F9B           CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F  01FA2           PUSHAB  CRLF_SHIFT
                                           06   DD  01FA8           PUSHL   #6
                                00000000G  EF   9F  01FAA           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01FB0           CALLS   #3,PAS$WRITE_STRING
                                FFFFAFBD   EF   9F  01FB7           PUSHAB  C.AIA
                                           24   DD  01FBD           PUSHL   #36
                                00000000G  EF   9F  01FBF           PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF           03  FB  01FC5           CALLS   #3,PAS$WRITE_STRING
                                      0000V  31   01FCC           BRW     169$
              06                00 00000108G  EF   CF  01FCF 102$:     CASEL   IDATA+264,#0,#6                 ; 1087
                                      0000V      01FD7               .DISPL  103$
                                      0000V      01FD9               .DISPL  103$
                                      0000V      01FDB               .DISPL  103$
                                      0000V      01FDD               .DISPL  104$
```

```
                                0000V     01FDF            .DISPL  105$
                                0000V     01FE1            .DISPL  103$
                                0000V     01FE3            .DISPL  103$
                                0000V  31 01FE5            BRW     106$
                    00000000G   EF    9F 01FE8 103$:     PUSHAB  SHIFT                                   ; 1093
                                04    DD 01FEE           PUSHL   #4
                    00000000G   EF    9F 01FF0           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 01FF6           CALLS   #3,PAS$WRITE_STRING
                    FFFFAF9B    EF    9F 01FFD           PUSHAB  C.AIB
                                10    DD 02003           PUSHL   #16
                    00000000G   EF    9F 02005           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 0200B           CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 02012           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01    FB 02018           CALLS   #1,PAS$WRITELN2
                                00V   11 0201F           BRB     107$
                    00000000G   EF    9F 02021 104$:     PUSHAB  SHIFT                                   ; 1094
                                04    DD 02027           PUSHL   #4
                    00000000G   EF    9F 02029           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 0202F           CALLS   #3,PAS$WRITE_STRING
                    FFFFAF72    EF    9F 02036           PUSHAB  C.AIC
                                14    DD 0203C           PUSHL   #20
                    00000000G   EF    9F 0203E           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 02044           CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 0204B           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01    FB 02051           CALLS   #1,PAS$WRITELN2
                                00V   11 02058           BRB     107$
                    00000000G   EF    9F 0205A 105$:     PUSHAB  SHIFT                                   ; 1095
                                04    DD 02060           PUSHL   #4
                    00000000G   EF    9F 02062           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 02068           CALLS   #3,PAS$WRITE_STRING
                    FFFFAF4D    EF    9F 0206F           PUSHAB  C.AID
                                2D    DD 02075           PUSHL   #45
                    00000000G   EF    9F 02077           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 0207D           CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 02084           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01    FB 0208A           CALLS   #1,PAS$WRITELN2
                                00V   11 02091           BRB     107$
                                      02093 106$:
                    00000000G   EF    9F 02093 107$:     PUSHAB  SHIFT                                   ; 1104
                                04    DD 02099           PUSHL   #4
                    00000000G   EF    9F 0209B           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 020A1           CALLS   #3,PAS$WRITE_STRING
                    FFFFAF44    EF    9F 020A8           PUSHAB  C.AIE
                                21    DD 020AE           PUSHL   #33
                    00000000G   EF    9F 020B0           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 020B6           CALLS   #3,PAS$WRITE_STRING
                                0000V 31 020BD           BRW     169$
                    00000000G   EF    9F 020C0 108$:     PUSHAB  SHIFT                                   ; 1111
                                04    DD 020C6           PUSHL   #4
                    00000000G   EF    9F 020C8           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 020CE           CALLS   #3,PAS$WRITE_STRING
                    FFFFAF3B    EF    9F 020D5           PUSHAB  C.AIF
                                17    DD 020DB           PUSHL   #23
                    00000000G   EF    9F 020DD           PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03    FB 020E3           CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 020EA           PUSHAB  CUR_MAX_FIXED
00000000G   EF                  01    FB 020F0           CALLS   #1,NUM_CEN
```

```
                                50    DD 020F7          PUSHL   R0
                          00000000G   EF    DD 020F9    PUSHL   CUR_MAX_FIXED
                          00000000G   EF    9F 020FF    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 02105    CALLS   #3,PAS$WRITE_INTEGER
                          FFFFAF1C    EF    9F 0210C    PUSHAB  C.AIG
                                07    DD 02112          PUSHL   #7
                          00000000G   EF    9F 02114    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 0211A    CALLS   #3,PAS$WRITE_STRING
                                0000V 31    02121       BRW     169$
                          00000000G   EF    9F 02124 109$:  PUSHAB  SHIFT                           ; 1118
                                04    DD 0212A          PUSHL   #4
                          00000000G   EF    9F 0212C    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 02132    CALLS   #3,PAS$WRITE_STRING
          00V00000000G    EF          00    E1 02139    BBC     #0,VARIABLE_RECORDS,111$        ; 1120
                          FFFFAEEF    EF    9F 02141    PUSHAB  C.AIH                           ; 1122
                                05    DD 02147          PUSHL   #5
                          00000000G   EF    9F 02149    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 0214F    CALLS   #3,PAS$WRITE_STRING
                          FFFFAEE2    EF    9F 02156 111$:  PUSHAB  C.AII                           ; 1124
                                0B    DD 0215C          PUSHL   #11
                          00000000G   EF    9F 0215E    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 02164    CALLS   #3,PAS$WRITE_STRING
                          0F 00000100G EF     D1 0216B    CMPL    IDATA+256,#15                   ; 1126
                                00V   12    02172       BNEQ    113$
                          FFFFAED0    EF    9F 02174    PUSHAB  C.AIJ                           ; 1128
                                06    DD 0217A          PUSHL   #6
                          00000000G   EF    9F 0217C    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 02182    CALLS   #3,PAS$WRITE_STRING
          00V00000000G    EF          00    E0 02189 113$:  BBS     #0,VARIABLE_RECORDS,115$        ; 1130
                                01    DD 02191          PUSHL   #1                              ; 1132
                          7E 00000000G EF     9A 02193    MOVZBL  TAB,-(SP)
                          00000000G   EF    9F 0219A    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 021A0    CALLS   #3,PAS$WRITE_CHAR
                          FFFFAEA5    EF    9F 021A7 115$:  PUSHAB  C.AIK                           ; 1134
                                05    DD 021AD          PUSHL   #5
                          00000000G   EF    9F 021AF    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 021B5    CALLS   #3,PAS$WRITE_STRING
                          00000000G   EF    9F 021BC    PUSHAB  CUR_MAX_REC
                00000000G EF          01    FB 021C2    CALLS   #1,NUM_CEN
                                50    DD 021C9          PUSHL   R0
                          00000000G   EF    DD 021CB    PUSHL   CUR_MAX_REC
                          00000000G   EF    9F 021D1    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 021D7    CALLS   #3,PAS$WRITE_INTEGER
                                01    DD 021DE          PUSHL   #1
                                29    DD 021E0          PUSHL   #41
                          00000000G   EF    9F 021E2    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 021E8    CALLS   #3,PAS$WRITE_CHAR
                          00000000G   EF    9F 021EF    PUSHAB  ANSI_REVERSE
                                04    DD 021F5          PUSHL   #4
                          00000000G   EF    9F 021F7    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 021FD    CALLS   #3,PAS$WRITE_STRING
                          FFFFAE50    EF    9F 02204    PUSHAB  C.AIL
                                03    DD 0220A          PUSHL   #3
                          00000000G   EF    9F 0220C    PUSHAB  PAS$FV_OUTPUT
                00000000G EF          03    FB 02212    CALLS   #3,PAS$WRITE_STRING
                          00000000G   EF    9F 02219    PUSHAB  ANSI_RESET
                                04    DD 0221F          PUSHL   #4
```

```
                 00000000G  EF    9F 02221          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02227          CALLS   #3,PAS$WRITE_STRING
                 FFFFAE2A   EF    9F 0222E          PUSHAB  C.AIM
                            03    DD 02234          PUSHL   #3
                 00000000G  EF    9F 02236          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 0223C          CALLS   #3,PAS$WRITE_STRING
                          0000V   31 02243          BRW     169$
                 00000000   8F    DF 02246 116$:    PUSHAL  #0                                      ; 1147
00000000G  EF               01    FB 0224C          CALLS   #1,CLEAR
00V00000000G  EF            00    E0 02253          BBS     #0,FULL_PROMPT,118$                     ; 1149
03 00000000G  EF            00    E0 0225B          BBS     #0,TEMP_FULL_PROMPT,..+3
                          0000V   31 02263          BRW     121$
                 00000000G  EF    9F 02266 118$:    PUSHAB  SHIFT                                   ; 1153
                            04    DD 0226C          PUSHL   #4
                 00000000G  EF    9F 0226E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02274          CALLS   #3,PAS$WRITE_STRING
                 FFFFADE1   EF    9F 0227B          PUSHAB  C.AIN
                            02    DD 02281          PUSHL   #2
                 00000000G  EF    9F 02283          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02289          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF    9F 02290          PUSHAB  ANSI_REVERSE
                            04    DD 02296          PUSHL   #4
                 00000000G  EF    9F 02298          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 0229E          CALLS   #3,PAS$WRITE_STRING
                 FFFFADBB   EF    9F 022A5          PUSHAB  C.AIO
                            04    DD 022AB          PUSHL   #4
                 00000000G  EF    9F 022AD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 022B3          CALLS   #3,PAS$WRITE_STRING
                            03    DD 022BA          PUSHL   #3
                 00000084G  EF    DD 022BC          PUSHL   IDATA+132
                 00000000G  EF    9F 022C2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 022C8          CALLS   #3,PAS$WRITE_INTEGER
                 FFFFAD95   EF    9F 022CF          PUSHAB  C.AIP
                            16    DD 022D5          PUSHL   #22
                 00000000G  EF    9F 022D7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 022DD          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF    9F 022E4          PUSHAB  ANSI_RESET
                            04    DD 022EA          PUSHL   #4
                 00000000G  EF    9F 022EC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 022F2          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF    9F 022F9          PUSHAB  CRLF
                            02    DD 022FF          PUSHL   #2
                 00000000G  EF    9F 02301          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02307          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF    9F 0230E          PUSHAB  CRLF_SHIFT
                            06    DD 02314          PUSHL   #6
                 00000000G  EF    9F 02316          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 0231C          CALLS   #3,PAS$WRITE_STRING
                 FFFFAD59   EF    9F 02323          PUSHAB  C.AIQ
                            37    DD 02329          PUSHL   #55
                 00000000G  EF    9F 0232B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02331          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF    9F 02338          PUSHAB  CRLF_SHIFT
                            06    DD 0233E          PUSHL   #6
                 00000000G  EF    9F 02340          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03    FB 02346          CALLS   #3,PAS$WRITE_STRING
                 FFFFAD67   EF    9F 0234D          PUSHAB  C.AIR
```

```
                    3C DD 02353          PUSHL   #60
                00000000G EF 9F 02355    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 0235B      CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 02362    PUSHAB  CRLF_SHIFT
                    06 DD 02368          PUSHL   #6
                00000000G EF 9F 0236A    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 02370      CALLS   #3,PAS$WRITE_STRING
                FFFFAD79 EF 9F 02377     PUSHAB  C.AIS
                    3B DD 0237D          PUSHL   #59
                00000000G EF 9F 0237F    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 02385      CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 0238C    PUSHAB  PAS$FV_OUTPUT
00000000G EF            01 FB 02392      CALLS   #1,PAS$WRITELN2
                00000084G EF D5 02399    TSTL    IDATA+132                       ; 1168
                    03 13 0239F          BEQL    .+3
                  0000V 31 023A1         BRW     120$
                00000000G EF 9F 023A4    PUSHAB  SHIFT                           ; 1172
                    04 DD 023AA          PUSHL   #4
                00000000G EF 9F 023AC    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 023B2      CALLS   #3,PAS$WRITE_STRING
                FFFFAD73 EF 9F 023B9     PUSHAB  C.AIT
                    3E DD 023BF          PUSHL   #62
                00000000G EF 9F 023C1    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 023C7      CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 023CE    PUSHAB  CRLF_SHIFT
                    06 DD 023D4          PUSHL   #6
                00000000G EF 9F 023D6    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 023DC      CALLS   #3,PAS$WRITE_STRING
                FFFFAD89 EF 9F 023E3     PUSHAB  C.AIU
                    3C DD 023E9          PUSHL   #60
                00000000G EF 9F 023EB    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 023F1      CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 023F8    PUSHAB  CRLF_SHIFT
                    06 DD 023FE          PUSHL   #6
                00000000G EF 9F 02400    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 02406      CALLS   #3,PAS$WRITE_STRING
                FFFFAD9B EF 9F 0240D     PUSHAB  C.AIV
                    3B DD 02413          PUSHL   #59
                00000000G EF 9F 02415    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 0241B      CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 02422    PUSHAB  PAS$FV_OUTPUT
00000000G EF            01 FB 02428      CALLS   #1,PAS$WRITELN2
                00000000G EF 9F 0242F 120$: PUSHAB PAS$FV_OUTPUT                ; 1181
00000000G EF            01 FB 02435      CALLS   #1,PAS$WRITELN2
                    00V 11 0243C         BRB     125$
                00000000G EF 9F 0243E 121$: PUSHAB SHIFT                        ; 1189
                    04 DD 02444          PUSHL   #4
                00000000G EF 9F 02446    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 0244C      CALLS   #3,PAS$WRITE_STRING
                FFFFAD91 EF 9F 02453     PUSHAB  C.AIW
                    0E DD 02459          PUSHL   #14
                00000000G EF 9F 0245B    PUSHAB  PAS$FV_OUTPUT
00000000G EF            03 FB 02461      CALLS   #3,PAS$WRITE_STRING
                00000084G EF D5 02468    TSTL    IDATA+132                       ; 1191
                    00V 12 0246E         BNEQ    123$
                FFFFAD84 EF 9F 02470     PUSHAB  C.AIX                           ; 1193
                    11 DD 02476          PUSHL   #17
```

Generated Code

```
                    00000000G   EF    9F 02478          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF        03 FB 0247E       CALLS   #3,PAS$WRITE_STRING
                                  00V   11 02485        BRB     124$
                                  01    DD 02487 123$:  PUSHL   #1                              ; 1197
                                  29    DD 02489        PUSHL   #41
                    00000000G   EF    9F 0248B          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF        03 FB 02491       CALLS   #3,PAS$WRITE_CHAR
                    00000000G   EF    9F 02498 124$:    PUSHAB  PAS$FV_OUTPUT                   ; 1199
                00000000G   EF        01 FB 0249E       CALLS   #1,PAS$WRITELN2
                    00000000G   EF    9F 024A5 125$:    PUSHAB  SHIFT                           ; 1206
                                  04    DD 024AB        PUSHL   #4
                    00000000G   EF    9F 024AD          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF        03 FB 024B3       CALLS   #3,PAS$WRITE_STRING
                        FFFFAD4E    EF    9F 024BA      PUSHAB  C.AIY
                                  21    DD 024C0        PUSHL   #33
                    00000000G   EF    9F 024C2          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF        03 FB 024C8       CALLS   #3,PAS$WRITE_STRING
              50 00000000G   EF        19 C5 024CF       MULL3   #25,QTAB_OFFSET,R0              ; 1208
50 FFFFFEF2GEF40                  20    00 EE 024D7       EXTV    #0,#32,QTAB-270[R0],R0
                05                00    50 CF 024E1       CASEL   R0,#0,#5
                                      0000V   024E5       .DISPL  127$
                                      0000V   024E7       .DISPL  129$
                                      0000V   024E9       .DISPL  130$
                                      0000V   024EB       .DISPL  131$
                                      0000V   024ED       .DISPL  128$
                                      0000V   024EF       .DISPL  126$
                                      0000V 31 024F1      BRW     132$
                        FFFFAD38    EF    9F 024F4 126$:  PUSHAB  C.AIZ                           ; 1210
                                  08    DD 024FA        PUSHL   #8
                    00000000G   EF    9F 024FC          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF        03 FB 02502       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02509    BRW     169$
                        FFFFAD28    EF    9F 0250C 127$:  PUSHAB  C.AJA                           ; 1211
                                  08    DD 02512        PUSHL   #8
                    00000000G   EF    9F 02514          PUSHAB  PAS$FV_OUTPUT
            00000000G   EF            03 FB 0251A       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02521    BRW     169$
                        FFFFAD18    EF    9F 02524 128$:  PUSHAB  C.AJB                           ; 1212
                                  07    DD 0252A        PUSHL   #7
                    00000000G   EF    9F 0252C          PUSHAB  PAS$FV_OUTPUT
            00000000G   EF            03 FB 02532       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02539    BRW     169$
                        FFFFAD08    EF    9F 0253C 129$:  PUSHAB  C.AJC                           ; 1213
                                  07    DD 02542        PUSHL   #7
                    00000000G   EF    9F 02544          PUSHAB  PAS$FV_OUTPUT
            00000000G   EF            03 FB 0254A       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02551    BRW     169$
                        FFFFACF8    EF    9F 02554 130$:  PUSHAB  C.AJD                           ; 1214
                                  08    DD 0255A        PUSHL   #8
                    00000000G   EF    9F 0255C          PUSHAB  PAS$FV_OUTPUT
            00000000G   EF            03 FB 02562       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02569    BRW     169$
                        FFFFACE8    EF    9F 0256C 131$:  PUSHAB  C.AJE                           ; 1215
                                  07    DD 02572        PUSHL   #7
                    00000000G   EF    9F 02574          PUSHAB  PAS$FV_OUTPUT
            00000000G   EF            03 FB 0257A       CALLS   #3,PAS$WRITE_STRING
                                      0000V 31 02581    BRW     169$
```

```
                    0000V  31 02584 132$:   BRW     169$
                00000000   8F   DF 02587 134$:   PUSHAL  #0                                      ; 1232
    00000000G   EF          01   FB 0258D         CALLS   #1,CLEAR
00V00000000G   EF          00   E0 02594         BBS     #0,FULL_PROMPT,136$                     ; 1234
03 00000000G   EF          00   E0 0259C         BBS     #0,TEMP_FULL_PROMPT,.+3
                    0000V  31 025A4             BRW     140$
                00000000G  EF   9F 025A7 136$:   PUSHAB  SHIFT                                   ; 1238
                           04   DD 025AD         PUSHL   #4
                00000000G  EF   9F 025AF         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 025B5         CALLS   #3,PAS$WRITE_STRING
                FFFFACA0   EF   9F 025BC         PUSHAB  C.AJF
                           02   DD 025C2         PUSHL   #2
                00000000G  EF   9F 025C4         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 025CA         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 025D1         PUSHAB  ANSI_REVERSE
                           04   DD 025D7         PUSHL   #4
                00000000G  EF   9F 025D9         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 025DF         CALLS   #3,PAS$WRITE_STRING
                FFFFAC7A   EF   9F 025E6         PUSHAB  C.AJG
                           1C   DD 025EC         PUSHL   #28
                00000000G  EF   9F 025EE         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 025F4         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 025FB         PUSHAB  ANSI_RESET
                           04   DD 02601         PUSHL   #4
                00000000G  EF   9F 02603         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 02609         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 02610         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          01   FB 02616         CALLS   #1,PAS$WRITELN2
03 00000000G   EF          00   E0 0261D         BBS     #0,DEC_CRT,.+3                          ; 1245
                    0000V  31 02625             BRW     138$
                00r00000G  EF   9F 02628         PUSHAB  CRLF                                    ; 1249
                           02   DD 0262E         PUSHL   #2
                00000000G  EF   9F 02630         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 02636         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0263D         PUSHAB  LOW_SHIFT
                           03   DD 02643         PUSHL   #3
                00000000G  EF   9F 02645         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 0264B         CALLS   #3,PAS$WRITE_STRING
                FFFFAC2A   EF   9F 02652         PUSHAB  C.AJH
                0000005A   8F   DD 02658         PUSHL   #90
                00000000G  EF   9F 0265E         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 02664         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0266B         PUSHAB  CRLF
                           02   DD 02671         PUSHL   #2
                00000000G  EF   9F 02673         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 02679         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 02680         PUSHAB  LOW_SHIFT
                           03   DD 02686         PUSHL   #3
                00000000G  EF   9F 02688         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 0268E         CALLS   #3,PAS$WRITE_STRING
                FFFFAC43   EF   9F 0269>         PUSHAB  C.AJI
                0000005F   8F   DD 0269B         PUSHL   #95
                00000000G  EF   9F 026A1         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF          03   FB 026A7         CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 026AE         PUSHAB  CRLF
                           02   DD 026B4         PUSHL   #2
                00000000G  EF   9F 026B6         PUSHAB  PAS$FV_OUTPUT
```

```
00000000G  EF                03  FB 026BC         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 026C3         PUSHAB  LOW_SHIFT
                             03  DD 026C9         PUSHL   #3
                00000000G    EF  9F 026CB         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 026D1         CALLS   #3,PAS$WRITE_STRING
                FFFFAC60     EF  9F 026D8         PUSHAB  C.AJJ
                00000059     8F  DD 026DE         PUSHL   #89
                00000000G    EF  9F 026E4         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 026EA         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 026F1         PUSHAB  CRLF
                             02  DD 026F7         PUSHL   #2
                00000000G    EF  9F 026F9         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 026FF         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 02706         PUSHAB  LOW_SHIFT
                             03  DD 0270C         PUSHL   #3
                00000000G    EF  9F 0270E         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 02714         CALLS   #3,PAS$WRITE_STRING
                FFFFAC79     EF  9F 0271B         PUSHAB  C.AJK
                0000005F     8F  DD 02721         PUSHL   #95
                00000000G    EF  9F 02727         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0272D         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 02734         PUSHAB  CRLF
                             02  DD 0273A         PUSHL   #2
                00000000G    EF  9F 0273C         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 02742         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 02749         PUSHAB  LOW_SHIFT
                             03  DD 0274F         PUSHL   #3
                00000000G    EF  9F 02751         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 02757         CALLS   #3,PAS$WRITE_STRING
                FFFFAC96     EF  9F 0275E         PUSHAB  C.AJL
                0000005B     8F  DD 02764         PUSHL   #91
                00000000G    EF  9F 0276A         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 02770         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 02777         PUSHAB  CRLF
                             02  DD 0277D         PUSHL   #2
                00000000G    EF  9F 0277F         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 02785         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 0278C         PUSHAB  LOW_SHIFT
                             03  DD 02792         PUSHL   #3
                00000000G    EF  9F 02794         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0279A         CALLS   #3,PAS$WRITE_STRING
                FFFFACAF     EF  9F 027A1         PUSHAB  C.AJM
                0000005F     8F  DD 027A7         PUSHL   #95
                00000000G    EF  9F 027AD         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 027B3         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 027BA         PUSHAB  CRLF
                             02  DD 027C0         PUSHL   #2
                00000000G    EF  9F 027C2         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 027C8         CALLS   #3,PAS$WRITE_STRING
                00000000G    EF  9F 027CF         PUSHAB  LOW_SHIFT
                             03  DD 027D5         PUSHL   #3
                00000000G    EF  9F 027D7         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 027DD         CALLS   #3,PAS$WRITE_STRING
                FFFFACCC     EF  9F 027E4         PUSHAB  C.AJN
                0000005D     8F  DD 027EA         PUSHL   #93
                00000000G    EF  9F 027F0         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 027F6         CALLS   #3,PAS$WRITE_STRING
```

```
                    00000000G   EF   9F 027FD          PUSHAB  CRLF
                                02   DD 02803          PUSHL   #2
                    00000000G   EF   9F 02805          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0280B          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02812          PUSHAB  LOW_SHIFT
                                03   DD 02818          PUSHL   #3
                    00000000G   EF   9F 0281A          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 02820          CALLS   #3,PAS$WRITE_STRING
                    FFFFACE9    EF   9F 02827          PUSHAB  C.AJO
                    0000005F    8F   DD 0282D          PUSHL   #95
                    00000000G   EF   9F 02833          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 02839          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02840          PUSHAB  CRLF
                                02   DD 02846          PUSHL   #2
                    00000000G   EF   9F 02848          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0284E          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02855          PUSHAB  LOW_SHIFT
                                03   DD 0285B          PUSHL   #3
                    00000000G   EF   9F 0285D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 02863          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD06    EF   9F 0286A          PUSHAB  C.AJP
                    00000057    8F   DD 02870          PUSHL   #87
                    00000000G   EF   9F 02876          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0287C          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02883          PUSHAB  CRLF
                                02   DD 02889          PUSHL   #2
                    00000000G   EF   9F 0288B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 02891          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02898          PUSHAB  LOW_SHIFT
                                03   DD 0289E          PUSHL   #3
                    00000000G   EF   9F 028A0          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 028A6          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD1B    EF   9F 028AD          PUSHAB  C.AJQ
                    0000004B    8F   DD 028B3          PUSHL   #75
                    00000000G   EF   9F 028B9          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 028BF          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 028C6          PUSHAB  CRLF
                                02   DD 028CC          PUSHL   #2
                    00000000G   EF   9F 028CE          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 028D4          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 028DB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01   FB 028E1          CALLS   #1,PAS$WRITELN2
                              0000V  31 028E8          BRW     139$
                    00000000G   EF   9F 028EB 138$:    PUSHAB  CRLF                    ; 1277
                                02   DD 028F1          PUSHL   #2
                    00000000G   EF   9F 028F3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 028F9          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 02900          PUSHAB  LOW_SHIFT
                                03   DD 02906          PUSHL   #3
                    00000000G   EF   9F 02908          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0290E          CALLS   #3,PAS$WRITE_STRING
                    FFFFACFF    EF   9F 02915          PUSHAB  C.AJR
                    0000004E    8F   DD 0291B          PUSHL   #78
                    00000000G   EF   9F 02921          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 02927          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 0292E          PUSHAB  CRLF
                                02   DD 02934          PUSHL   #2
```

```
                 00000000G   EF   9F 02936         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 0293C         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02943         PUSHAB  LOW_SHIFT
                               03   DD 02949         PUSHL   #3
                 00000000G   EF   9F 0294B         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02951         CALLS   #3,PAS$WRITE_STRING
                 FFFFAD0C    EF   9F 02958         PUSHAB  C.AJS
                 0000004E    8F   DD 0295E         PUSHL   #78
                 00000000G   EF   9F 02964         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 0296A         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02971         PUSHAB  CRLF
                               02   DD 02977         PUSHL   #2
                 00000000G   EF   9F 02979         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 0297F         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02986         PUSHAB  LOW_SHIFT
                               03   DD 0298C         PUSHL   #3
                 00000000G   EF   9F 0298E         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02994         CALLS   #3,PAS$WRITE_STRING
                 FFFFAD19    EF   9F 0299B         PUSHAB  C.AJT
                 0000004E    8F   DD 029A1         PUSHL   #78
                 00000000G   EF   9F 029A7         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 029AD         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 029B4         PUSHAB  CRLF
                               02   DD 029BA         PUSHL   #2
                 00000000G   EF   9F 029BC         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 029C2         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 029C9         PUSHAB  LOW_SHIFT
                               03   DD 029CF         PUSHL   #3
                 00000000G   EF   9F 029D1         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 029D7         CALLS   #3,PAS$WRITE_STRING
                 FFFFAD26    EF   9F 029DE         PUSHAB  C.AJU
                 0000004E    8F   DD 029E4         PUSHL   #78
                 00000000G   EF   9F 029EA         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 029F0         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 029F7         PUSHAB  CRLF
                               02   DD 029FD         PUSHL   #2
                 00000000G   EF   9F 029FF         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02A05         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02A0C         PUSHAB  LOW_SHIFT
                               03   DD 02A12         PUSHL   #3
                 00000000G   EF   9F 02A14         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02A1A         CALLS   #3,PAS$WRITE_STRING
                 FFFFAD33    EF   9F 02A21         PUSHAB  C.AJV
                 0000004E    8F   DD 02A27         PUSHL   #78
                 00000000G   EF   9F 02A2D         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02A33         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02A3A         PUSHAB  CRLF
                               02   DD 02A40         PUSHL   #2
                 00000000G   EF   9F 02A42         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02A48         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F 02A4F         PUSHAB  LOW_SHIFT
                               03   DD 02A55         PUSHL   #3
                 00000000G   EF   9F 02A57         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                03   FB 02A5D         CALLS   #3,PAS$WRITE_STRING
                 FFFFAD40    EF   9F 02A64         PUSHAB  C.AJW
                 0000004E    8F   DD 02A6A         PUSHL   #78
                 00000000G   EF   9F 02A70         PUSHAB  PAS$FV_OUTPUT
```

Generated Code

```
00000000G  EF                    03  FB 02A76          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02A7D          PUSHAB  CRLF
                                 02  DD 02A83          PUSHL   #2
                    00000000G    EF  9F 02A85          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02A8B          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02A92          PUSHAB  LOW_SHIFT
                                 03  DD 02A98          PUSHL   #3
                    00000000G    EF  9F 02A9A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02AA0          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD4D     EF  9F 02AA7          PUSHAB  C.AJX
                    0000004E     8F  DD 02AAD          PUSHL   #78
                    00000000G    EF  9F 02AB3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02AB9          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02AC0          PUSHAB  CRLF
                                 02  DD 02AC6          PUSHL   #2
                    00000000G    EF  9F 02AC8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02ACE          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02AD5          PUSHAB  LOW_SHIFT
                                 03  DD 02ADB          PUSHL   #3
                    00000000G    EF  9F 02ADD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02AE3          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD5A     EF  9F 02AEA          PUSHAB  C.AJY
                    0000004E     8F  DD 02AF0          PUSHL   #78
                    00000000G    EF  9F 02AF6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02AFC          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B03          PUSHAB  CRLF
                                 02  DD 02B09          PUSHL   #2
                    00000000G    EF  9F 02B0B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B11          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B18          PUSHAB  LOW_SHIFT
                                 03  DD 02B1E          PUSHL   #3
                    00000000G    EF  9F 02B20          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B26          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD67     EF  9F 02B2D          PUSHAB  C.AJZ
                    0000004E     8F  DD 02B33          PUSHL   #78
                    00000000G    EF  9F 02B39          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B3F          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B46          PUSHAB  CRLF
                                 02  DD 02B4C          PUSHL   #2
                    00000000G    EF  9F 02B4E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B54          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B5B          PUSHAB  LOW_SHIFT
                                 03  DD 02B61          PUSHL   #3
                    00000000G    EF  9F 02B63          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B69          CALLS   #3,PAS$WRITE_STRING
                    FFFFAD74     EF  9F 02B70          PUSHAB  C.AKA
                    0000004B     8F  DD 02B76          PUSHL   #75
                    00000000G    EF  9F 02B7C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B82          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B89          PUSHAB  CRLF
                                 02  DD 02B8F          PUSHL   #2
                    00000000G    EF  9F 02B91          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03  FB 02B97          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF  9F 02B9E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    01  FB 02BA4          CALLS   #1,PAS$WRITELN2
                                 00V 11 02BAB 139$:    BRB     141$                    ;
                    00000000G    EF  9F 02BAD 140$:    PUSHAB  SHIFT                   ; 1307
```

```
                                 04 DD 02BB3          PUSHL   #4
                      00000000G  EF 9F 02BB5          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02BBB          CALLS   #3,PAS$WRITE_STRING
                           FFFFAD6E EF 9F 02BC2       PUSHAB  C.AKB
                                 1B DD 02BC8          PUSHL   #27
                      00000000G  EF 9F 02BCA          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02BD0          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF 9F 02BD7          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    01 FB 02BDD          CALLS   #1,PAS$WRITELN2
                      00000000G  EF 9F 02BE4 141$:    PUSHAB  SHIFT                           ; 1314
                                 04 DD 02BEA          PUSHL   #4
                      00000000G  EF 9F 02BEC          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02BF2          CALLS   #3,PAS$WRITE_STRING
                           FFFFAD53 EF 9F 02BF9       PUSHAB  C.AKC
                                 2B DD 02BFF          PUSHL   #43
                      00000000G  EF 9F 02C01          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02C07          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF 9F 02C0E          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    01 FB 02C14          CALLS   #1,PAS$WRITELN2
                      00000000G  EF 9F 02C1B          PUSHAB  SHIFT                           ; 1315
                                 04 DD 02C21          PUSHL   #4
                      00000000G  EF 9F 02C23          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02C29          CALLS   #3,PAS$WRITE_STRING
                           FFFFAD48 EF 9F 02C30       PUSHAB  C.AKD
                                 2F DD 02C36          PUSHL   #47
                      00000000G  EF 9F 02C38          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02C3E          CALLS   #3,PAS$WRITE_STRING
                           0000V 31 02C45             BRW     169$
                      00000000    8F DF 02C48 142$:    PUSHAL  #0                            ; 1327
                   00000000G  EF    01 FB 02C4E       CALLS   #1,CLEAR
                00V000000000G  EF    00 E0 02C55       BBS     #0,FULL_PROMPT,144$           ; 1329
                03 000000000G  EF    00 E0 02C5D       BBS     #0,TEMP_FULL_PROMPT,,+3
                           0000V 31 02C65             BRW     145$
                      00000000G  EF 9F 02C68 144$:    PUSHAB  SHIFT                           ; 1333
                                 04 DD 02C6E          PUSHL   #4
                      00000000G  EF 9F 02C70          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02C76          CALLS   #3,PAS$WRITE_STRING
                           FFFFAD2B EF 9F 02C7D       PUSHAB  C.AKE
                                 02 DD 02C83          PUSHL   #2
                      00000000G  EF 9F 02C85          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02C8B          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF 9F 02C92          PUSHAB  ANSI_REVERSE
                                 04 DD 02C98          PUSHL   #4
                      00000000G  EF 9F 02C9A          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02CA0          CALLS   #3,PAS$WRITE_STRING
                           FFFFAD05 EF 9F 02CA7       PUSHAB  C.AKF
                                 19 DD 02CAD          PUSHL   #25
                      00000000G  EF 9F 02CAF          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02CB5          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF 9F 02CBC          PUSHAB  ANSI_RESET
                                 04 DD 02CC2          PUSHL   #4
                      00000000G  EF 9F 02CC4          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02CCA          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF 9F 02CD1          PUSHAB  CRLF
                                 02 DD 02CD7          PUSHL   #2
                      00000000G  EF 9F 02CD9          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF    03 FB 02CDF          CALLS   #3,PAS$WRITE_STRING
```

```
                 00000000G  EF  9F  02CE6         PUSHAB  CRLF_SHIFT
                            06  DD  02CEC         PUSHL   #6
                 00000000G  EF  9F  02CEE         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02CF4         CALLS   #3,PAS$WRITE_STRING
                 FFFFACCD   EF  9F  02CFB         PUSHAB  C.AKG
                            26  DD  02D01         PUSHL   #38
                 00000000G  EF  9F  02D03         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D09         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02D10         PUSHAB  CRLF_SHIFT
                            06  DD  02D16         PUSHL   #6
                 00000000G  EF  9F  02D18         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D1E         CALLS   #3,PAS$WRITE_STRING
                 FFFFACCB   EF  9F  02D25         PUSHAB  C.AKH
                            21  DD  02D2B         PUSHL   #33
                 00000000G  EF  9F  02D2D         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D33         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02D3A         PUSHAB  CRLF_SHIFT
                            06  DD  02D40         PUSHL   #6
                 00000000G  EF  9F  02D42         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D48         CALLS   #3,PAS$WRITE_STRING
                 FFFFACC5   EF  9F  02D4F         PUSHAB  C.AKI
                            2B  DD  02D55         PUSHL   #43
                 00000000G  EF  9F  02D57         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D5D         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02D64         PUSHAB  CRLF_SHIFT
                            06  DD  02D6A         PUSHL   #6
                 00000000G  EF  9F  02D6C         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D72         CALLS   #3,PAS$WRITE_STRING
                 FFFFACC7   EF  9F  02D79         PUSHAB  C.AKJ
                            2C  DD  02D7F         PUSHL   #44
                 00000000G  EF  9F  02D81         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D87         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02D8E         PUSHAB  CRLF_SHIFT
                            06  DD  02D94         PUSHL   #6
                 00000000G  EF  9F  02D96         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02D9C         CALLS   #3,PAS$WRITE_STRING
                 FFFFACC9   EF  9F  02DA3         PUSHAB  C.AKK
                            2B  DD  02DA9         PUSHL   #43
                 00000000G  EF  9F  02DAB         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02DB1         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02DB8         PUSHAB  CRLF_SHIFT
                            06  DD  02DBE         PUSHL   #6
                 00000000G  EF  9F  02DC0         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02DC6         CALLS   #3,PAS$WRITE_STRING
                 FFFFACCB   EF  9F  02DCD         PUSHAB  C.AKL
                            23  DD  02DD3         PUSHL   #35
                 00000000G  EF  9F  02DD5         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02DDB         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02DE2         PUSHAB  CRLF_SHIFT
                            06  DD  02DE8         PUSHL   #6
                 00000000G  EF  9F  02DEA         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02DF0         CALLS   #3,PAS$WRITE_STRING
                 FFFFACC5   EF  9F  02DF7         PUSHAB  C.AKM
                            2A  DD  02DFD         PUSHL   #42
                 00000000G  EF  9F  02DFF         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03  FB  02E05         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02E0C         PUSHAB  CRLF_SHIFT
```

```
                                 06   DD 02E12          PUSHL   #6
                    00000000G    EF   9F 02E14          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E1A          CALLS   #3,PAS$WRITE_STRING
                    FFFFACC7     EF   9F 02E21          PUSHAB  C.AKN
                                 2F   DD 02E27          PUSHL   #47
                    00000000G    EF   9F 02E29          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E2F          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02E36          PUSHAB  CRLF
                                 02   DD 02E3C          PUSHL   #2
                    00000000G    EF   9F 02E3E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E44          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02E4B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 02E51          CALLS   #1,PAS$WRITELN2
                                 00V  11 02E58          BRB     146$
                    00000000G    EF   9F 02E5A 145$:    PUSHAB  SHIFT                           ; 1364
                                 04   DD 02E60          PUSHL   #4
                    00000000G    EF   9F 02E62          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E68          CALLS   #3,PAS$WRITE_STRING
                    FFFFACA9     EF   9F 02E6F          PUSHAB  C.AKO
                                 26   DD 02E75          PUSHL   #38
                    00000000G    EF   9F 02E77          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E7D          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02E84          PUSHAB  CRLF_SHIFT
                                 06   DD 02E8A          PUSHL   #6
                    00000000G    EF   9F 02E8C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02E92          CALLS   #3,PAS$WRITE_STRING
                    FFFFACA7     EF   9F 02E99          PUSHAB  C.AKP
                                 28   DD 02E9F          PUSHL   #40
                    00000000G    EF   9F 02EA1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02EA7          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02EAE          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 02EB4          CALLS   #1,PAS$WRITELN2
                    00000000G    EF   9F 02EBB 146$:    PUSHAB  SHIFT                           ; 1373
                                 04   DD 02EC1          PUSHL   #4
                    00000000G    EF   9F 02EC3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02EC9          CALLS   #3,PAS$WRITE_STRING
                    FFFFAC98     EF   9F 02ED0          PUSHAB  C.AKQ
                                 26   DD 02ED6          PUSHL   #38
                    00000000G    EF   9F 02ED8          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02EDE          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02EE5          PUSHAB  ANSI_REVERSE
                                 04   DD 02EEB          PUSHL   #4
                    00000000G    EF   9F 02EED          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02EF3          CALLS   #3,PAS$WRITE_STRING
                    FFFFAC96     EF   9F 02EFA          PUSHAB  C.AKR
                                 03   DD 02F00          PUSHL   #3
                    00000000G    EF   9F 02F02          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02F08          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 02F0F          PUSHAB  ANSI_RESET
                                 04   DD 02F15          PUSHL   #4
                    00000000G    EF   9F 02F17          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02F1D          CALLS   #3,PAS$WRITE_STRING
                    FFFFAC70     EF   9F 02F24          PUSHAB  C.AKS
                                 03   DD 02F2A          PUSHL   #3
                    00000000G    EF   9F 02F2C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 02F32          CALLS   #3,PAS$WRITE_STRING
                              0000V  31 02F39          BRW     169$
```

```
                 00000000G  EF   9F 02F3C 147$:   PUSHAB  SHIFT                          ; 1383
                            04   DD 02F42         PUSHL   #4
                 00000000G  EF   9F 02F44         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02F4A         CALLS   #3,PAS$WRITE_STRING
                 FFFFAC47   EF   9F 02F51         PUSHAB  C.AKT
                            16   DD 02F57         PUSHL   #22
                 00000000G  EF   9F 02F59         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02F5F         CALLS   #3,PAS$WRITE_STRING
              03 00000108G  EF   D1 02F66         CMPL    IDATA+264,#3                   ; 1385
                            03   13 02F6D         BEQL    .+3
                      0000V 31   31 02F6F         BRW     149$
                 00000000G  EF   D4 02F72         CLRL    EXTRA                          ; 1389
                 00000000G  EF   9F 02F78         PUSHAB  LOWMAX                         ; 1390
00000000G  EF               01   FB 02F7E         CALLS   #1,NUM_LEN
                            50   DD 02F85         PUSHL   R0
                 00000000G  EF   DD 02F87         PUSHL   LOWMAX
                 00000000G  EF   9F 02F8D         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02F93         CALLS   #3,PAS$WRITE_INTEGER
                            01   DD 02F9A         PUSHL   #1
                            2D   DD 02F9C         PUSHL   #45
                 00000000G  EF   9F 02F9E         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02FA4         CALLS   #3,PAS$WRITE_CHAR
                 00000000G  EF   9F 02FAB         PUSHAB  CUR_MAX_REC
00000000G  EF               01   FB 02FB1         CALLS   #1,NUM_LEN
                            50   DD 02FB8         PUSHL   R0
                 00000000G  EF   DD 02FBA         PUSHL   CUR_MAX_REC
                 00000000G  EF   9F 02FC0         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02FC6         CALLS   #3,PAS$WRITE_INTEGER
                            01   DD 02FCD         PUSHL   #1
                            29   DD 02FCF         PUSHL   #41
                 00000000G  EF   9F 02FD1         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02FD7         CALLS   #3,PAS$WRITE_CHAR
                 00000000G  EF   9F 02FDE         PUSHAB  ANSI_REVERSE
                            04   DD 02FE4         PUSHL   #4
                 00000000G  EF   9F 02FE6         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02FEC         CALLS   #3,PAS$WRITE_STRING
                 FFFFABBD   EF   9F 02FF3         PUSHAB  C.AKU
                            03   DD 02FF9         PUSHL   #3
                 00000000G  EF   9F 02FFB         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 03001         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 03008         PUSHAB  ANSI_RESET
                            04   DD 0300E         PUSHL   #4
                 00000000G  EF   9F 03010         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 03016         CALLS   #3,PAS$WRITE_STRING
50 00000000G  EF            19   C5 0301D         MULL3   #25,QTAB_OFFSET,R0             ; 1394
                 FFFFFEF1GEF40  94 03025         CLRB    QTAB-271[R0]
                      0000V 31   31 0302C         BRW     150$
00000000G  EF               02   D0 0302F 149$:   MOVL    #2,EXTRA                       ; 1402
                 FFFFAB7E   EF   9F 03036         PUSHAB  C.AKV                          ; 1403
                            02   DD 0303C         PUSHL   #2
                 00000000G  EF   9F 0303E         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 03044         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 0304B         PUSHAB  LOWMAX
00000000G  EF               01   FB 03051         CALLS   #1,NUM_LEN
                            50   DD 03058         PUSHL   R0
                 00000000G  EF   DD 0305A         PUSHL   LOWMAX
                 00000000G  EF   9F 03060         PUSHAB  PAS$FV_OUTPUT
```

```
                00000000G  EF                03   FB 03066           CALLS   #3,PAS$WRITE_INTEGER
                                             01   DD 0306D           PUSHL   #1
                                             2D   DD 0306F           PUSHL   #45
                           00000000G         EF   9F 03071           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 03077           CALLS   #3,PAS$WRITE_CHAR
                           00000000G         EF   9F 0307E           PUSHAB  CUR_MAX_REC
                00000000G  EF                01   FB 03084           CALLS   #1,NUM_LEN
                                             50   DD 0308B           PUSHL   R0
                           00000000G         EF   DD 0308D           PUSHL   CUR_MAX_REC
                           00000000G         EF   9F 03093           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 03099           CALLS   #3,PAS$WRITE_INTEGER
                           FFFFAB18          EF   9F 030A0           PUSHAB  C.AKW
                                             04   DD 030A6           PUSHL   #4
                           00000000G         EF   9F 030A8           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 030AE           CALLS   #3,PAS$WRITE_STRING
             50 00000000G  EF                19   C5 030B5           MULL3   #25,QTAB_OFFSET,R0                          ; 1406
                FFFFFEF1GEF40                01   90 030BD           MOVB    #1,QTAB-271[R0]
             50 00000000G  EF                19   C5 030C5           MULL3   #25,QTAB_OFFSET,R0                          ; 1407
                           5C FFFFFEF2G      EF   9E 030CD           MOVAB   QTAB-270,R12
                           5C                50   C0 030D4           ADDL2   R0,R12
                                             6C   D4 030D7           CLRL    (R12)
                           00000000G         EF   9F 030D9 150$:     PUSHAB  CUR_MAX_REC                                 ; 1411
                00000000G  EF                01   FB 030DF           CALLS   #1,NUM_LEN
                           5C                50   D0 030E6           MOVL    R0,R12
                           00000000G         EF   9F 030E9           PUSHAB  LOWMAX
                00000000G  EF                01   FB 030EF           CALLS   #1,NUM_LEN
                           50 00000000G      EF   C0 030F6           ADDL2   EXTRA,R0
                           50                5C   C0 030FD           ADDL2   R12,R0
                           09                50   D1 03100           CMPL    R0,#9
                                             00V  15 03103           BLEQ    152$
                           FFFFAAB7          EF   9F 03105           PUSHAB  C.AKX                                       ; 1415
                                             03   DD 0310B           PUSHL   #3
                           00000000G         EF   9F 0310D           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 03113           CALLS   #3,PAS$WRITE_STRING
                                        0000V  31   0311A           BRW     169$
                           FFFFAAA3          EF   9F 0311D 152$:     PUSHAB  C.AKY                                       ; 1419
                                             03   DD 03123           PUSHL   #3
                           00000000G         EF   9F 03125           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 0312B           CALLS   #3,PAS$WRITE_STRING
                                        0000V  31   03132           BRW     169$
                           00000000          8F   DF 03135 154$:     PUSHAL  #0                                          ; 1427
                00000000G  EF                01   FB 0313B           CALLS   #1,CLEAR
             03 00000000G  EF                00   E0 03142           BBS     #0,FULL_CHOICE,.+3                          ; 1429
                                        0000V  31   0314A           BRW     160$
             00V00000000G  EF                00   E0 0314D           BBS     #0,FULL_PROMPT,157$                         ; 1433
             03 00000000G  EF                00   E0 03155           BBS     #0,TEMP_FULL_PROMPT,.+3
                                        0000V  31   0315D           BRW     158$
                           00000000G         EF   9F 03160 157$:     PUSHAB  SHIFT                                       ; 1437
                                             04   DD 03166           PUSHL   #4
                           00000000G         EF   9F 03168           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 0316E           CALLS   #3,PAS$WRITE_STRING
                           FFFFAA4F          EF   9F 03175           PUSHAB  C.AKZ
                                             02   DD 0317B           PUSHL   #2
                           00000000G         EF   9F 0317D           PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 03183           CALLS   #3,PAS$WRITE_STRING
                           00000000G         EF   9F 0318A           PUSHAB  ANSI_REVERSE
                                             04   DD 03190           PUSHL   #4
```

```
                    00000000G   EF   9F 03192          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03198          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA29    EF   9F 0319F          PUSHAB  C.ALA
                                1A   DD 031A5          PUSHL   #26
                    00000000G   EF   9F 031A7          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 031AD          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 031B4          PUSHAB  ANSI_RESET
                                04   DD 031BA          PUSHL   #4
                    00000000G   EF   9F 031BC          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 031C2          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 031C9          PUSHAB  CRLF
                                02   DD 031CF          PUSHL   #2
                    00000000G   EF   9F 031D1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 031D7          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 031DE          PUSHAB  CRLF_SHIFT
                                06   DD 031E4          PUSHL   #6
                    00000000G   EF   9F 031E6          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 031EC          CALLS   #3,PAS$WRITE_STRING
                    FFFFA9F1    EF   9F 031F3          PUSHAB  C.ALB
                                3A   DD 031F9          PUSHL   #58
                    00000000G   EF   9F 031FB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03201          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 03208          PUSHAB  CRLF_SHIFT
                                06   DD 0320E          PUSHL   #6
                    00000000G   EF   9F 03210          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03216          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA03    EF   9F 0321D          PUSHAB  C.ALC
                                3B   DD 03223          PUSHL   #59
                    00000000G   EF   9F 03225          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0322B          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 03232          PUSHAB  CRLF_SHIFT
                                06   DD 03238          PUSHL   #6
                    00000000G   EF   9F 0323A          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03240          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA15    EF   9F 03247          PUSHAB  C.ALD
                                33   DD 0324D          PUSHL   #51
                    00000000G   EF   9F 0324F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03255          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 0325C          PUSHAB  CRLF_SHIFT
                                06   DD 03262          PUSHL   #6
                    00000000G   EF   9F 03264          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0326A          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA1F    EF   9F 03271          PUSHAB  C.ALE
                                33   DD 03277          PUSHL   #51
                    00000000G   EF   9F 03279          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 0327F          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 03286          PUSHAB  CRLF_SHIFT
                                06   DD 0328C          PUSHL   #6
                    00000000G   EF   9F 0328E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 03294          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA29    EF   9F 0329B          PUSHAB  C.ALF
                                2F   DD 032A1          PUSHL   #47
                    00000000G   EF   9F 032A3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 032A9          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 032B0          PUSHAB  CRLF_SHIFT
                                06   DD 032B6          PUSHL   #6
                    00000000G   EF   9F 032B8          PUSHAB  PAS$FV_OUTPUT
```

```
                        Generated Code

                                    03 FB 032BE          CALLS   #3,PAS$WRITE_STRING
                            FFFFAA2F EF 9F 032C5         PUSHAB  C.ALG
                                    3C DD 032CB          PUSHL   #60
                            00000000G EF 9F 032CD        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 032D3         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 032DA        PUSHAB  CRLF_SHIFT
                                    06 DD 032E0          PUSHL   #6
                            00000000G EF 9F 032E2        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 032E8         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA41 EF 9F 032EF         PUSHAB  C.ALH
                                    34 DD 032F5          PUSHL   #52
                            00000000G EF 9F 032F7        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 032FD         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 03304        PUSHAB  CRLF_SHIFT
                                    06 DD 0330A          PUSHL   #6
                            00000000G EF 9F 0330C        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 03312         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA4B EF 9F 03319         PUSHAB  C.ALI
                                    38 DD 0331F          PUSHL   #56
                            00000000G EF 9F 03321        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 03327         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 0332E        PUSHAB  CRLF_SHIFT
                                    06 DD 03334          PUSHL   #6
                            00000000G EF 9F 03336        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 0333C         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA59 EF 9F 03343         PUSHAB  C.ALJ
                                    3C DD 03349          PUSHL   #60
                            00000000G EF 9F 0334B        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 03351         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 03358        PUSHAB  CRLF_SHIFT
                                    06 DD 0335E          PUSHL   #6
                            00000000G EF 9F 03360        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 03366         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA6B EF 9F 0336D         PUSHAB  C.ALK
                                    3A DD 03373          PUSHL   #58
                            00000000G EF 9F 03375        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 0337B         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 03382        PUSHAB  CRLF_SHIFT
                                    06 DD 03388          PUSHL   #6
                            00000000G EF 9F 0338A        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 03390         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA7D EF 9F 03397         PUSHAB  C.ALL
                                    29 DD 0339D          PUSHL   #41
                            00000000G EF 9F 0339F        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 033A5         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 033AC        PUSHAB  CRLF
                                    02 DD 033B2          PUSHL   #2
                            00000000G EF 9F 033B4        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 033BA         CALLS   #3,PAS$WRITE_STRING
                            00000000G EF 9F 033C1        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 01 FB 033C7         CALLS   #1,PAS$WRITELN2
                                    00V 11 033CE         BRB     159$
                            00000000G EF 9F 033D0 158$:  PUSHAB  SHIFT                    ; 1474
                                    04 DD 033D6          PUSHL   #4
                            00000000G EF 9F 033D8        PUSHAB  PAS$FV_OUTPUT
                        00000000G EF 03 FB 033DE         CALLS   #3,PAS$WRITE_STRING
                            FFFFAA5B EF 9F 033E5         PUSHAB  C.ALM
```

```
                                 26   DD 033EB          PUSHL   #38
                    00000000G    EF   9F 033ED          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 033F3          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 033FA          PUSHAB  CRLF_SHIFT
                                 06   DD 03400          PUSHL   #6
                    00000000G    EF   9F 03402          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 03408          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA59     EF   9F 0340F          PUSHAB  C.ALN
                                 21   DD 03415          PUSHL   #33
                    00000000G    EF   9F 03417          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0341D          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 03424          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 0342A          CALLS   #1,PAS$WRITELN2
                           0000V 31   03431 159$:       BRW     165$
00V00000000G EF                  00   E0 03434 160$:    BBS     #0,FULL_PROMPT,162$                          ; 1487
03 00000000G EF                  00   E0 0343C          BBS     #0,TEMP_FULL_PROMPT,.+3
                           0000V 31   03444             BRW     163$
                    00000000G    EF   9F 03447 162$:    PUSHAB  SHIFT                                        ; 1491
                                 04   DD 0344D          PUSHL   #4
                    00000000G    EF   9F 0344F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 03455          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA2E     EF   9F 0345C          PUSHAB  C.ALO
                                 02   DD 03462          PUSHL   #2
                    00000000G    EF   9F 03464          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0346A          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 03471          PUSHAB  ANSI_REVERSE
                                 04   DD 03477          PUSHL   #4
                    00000000G    EF   9F 03479          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0347F          CALLS   #3,PAS$WRITE_STRING
                    FFFFAA06     EF   9F 03486          PUSHAB  C.ALP
                                 1C   DD 0348C          PUSHL   #28
                    00000000G    EF   9F 0348E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 03494          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 0349B          PUSHAB  ANSI_RESET
                                 04   DD 034A1          PUSHL   #4
                    00000000G    EF   9F 034A3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 034A9          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 034B0          PUSHAB  CRLF
                                 02   DD 034B6          PUSHL   #2
                    00000000G    EF   9F 034B8          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 034BE          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 034C5          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 034CB          CALLS   #1,PAS$WRITELN2
                    000000FC     8F   DD 034D2          PUSHL   #252                                         ; 1502
                                 07   DD 034D8          PUSHL   #7
                                 04   DD 034DA          PUSHL   #4
                    00000000G    EF   9F 034DC          PUSHAB  SYS$OUTPUT_NAME
                                 0B   DD 034E2          PUSHL   #11
                                 01   DD 034E4          PUSHL   #1
                    00000000G    EF   9F 034E6          PUSHAB  FDL_DEST
00000000G   EF                   07   FB 034EC          CALLS   #7,PAS$OPEN2
                    00000000G    EF   9F 034F3          PUSHAB  FDL_DEST                                     ; 1504
00000000G   EF                   01   FB 034F9          CALLS   #1,PAS$REWRITE2
00000000G   EF                   00   FB 03500          CALLS   #0,SHOW_ALL_PRIMARIES                        ; 1506
                    00000000G    EF   9F 03507          PUSHAB  FDL_DEST                                     ; 1508
00000000G   EF                   01   FB 0350D          CALLS   #1,PAS$CLOSE2
                                 00V  11 03514          BRB     165$
```

```
                                00000000G   EF    9F 03516 163$:   PUSHAB  SHIFT                                   ; 1514
                                            04    DD 0351C         PUSHL   #4
                                00000000G   EF    9F 0351E         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 03524         CALLS   #3,PAS$WRITE_STRING
                                FFFFA97D    EF    9F 0352B         PUSHAB  C.ALQ
                                            34    DD 03531         PUSHL   #52
                                00000000G   EF    9F 03533         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 03539         CALLS   #3,PAS$WRITE_STRING
                                00000000G   EF    9F 03540         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          01    FB 03546         CALLS   #1,PAS$WRITELN2
                                00000000G   EF    9F 0354D 165$:   PUSHAB  SHIFT                                   ; 1522
                                            04    DD 03553         PUSHL   #4
                                00000000G   EF    9F 03555         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 0355B         CALLS   #3,PAS$WRITE_STRING
                                FFFFA97A    EF    9F 03562         PUSHAB  C.ALR
                                            21    DD 03568         PUSHL   #33
                                00000000G   EF    9F 0356A         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 03570         CALLS   #3,PAS$WRITE_STRING
                                50 00000000G EF   9A 03577         MOVZBL  DEFAULT_PRIMARY,R0
                                7E 00000000GEF40  9A 0357E         MOVZBL  PRIMARY_WIDTH[R0],-(SP)
                                7E 00000000G EF   9A 03586         MOVZBL  DEFAULT_PRIMARY,-(SP)
                                FFFFA973    EF    9F 0358D         PUSHAB  C.ALS
                                00000000G   EF    9F 03593         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          04    FB 03599         CALLS   #4,PAS$WRITE_ENUMERATED
                                50 00000000G EF   9A 035A0         MOVZBL  DEFAULT_PRIMARY,R0                      ; 1525
                                10                50    D1 035A7         CMPL    R0,#16
                                                  00V   1E 035AA         BGEQU   167$
                00VFFFFAA2A     EF                50    E1 035AC         BBC     R0,C.ALT,167$
                                                  01    DD 035B4         PUSHL   #1                                ; 1527
                                                  20    DD 035B6         PUSHL   #32
                                00000000G   EF    9F 035B8         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 035BE         CALLS   #3,PAS$WRITE_CHAR
                                00000000G   EF    9F 035C5         PUSHAB  DEFAULT_PRINUM
                    00000000G   EF          01    FB 035CB         CALLS   #1,NUM_LEN
                                            50    DD 035D2         PUSHL   R0
                                00000000G   EF    DD 035D4         PUSHL   DEFAULT_PRINUM
                                00000000G   EF    9F 035DA         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 035E0         CALLS   #3,PAS$WRITE_INTEGER
                                FFFFA9F5    EF    9F 035E7 167$:   PUSHAB  C.ALU                                   ; 1529
                                            04    DD 035ED         PUSHL   #4
                                00000000G   EF    9F 035EF         PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF          03    FB 035F5         CALLS   #3,PAS$WRITE_STRING
                                                  00V   11 035FC         BRB     169$
                                                           035FE 168$:
                                                        04 035FE 169$:   RET                                       ; 1539

; Routine Size: 13823 bytes,    Routine Base: $CODE + 049C2

                                                        00000 SPREAD_LOW_HIGH:                                     ; 1585
                                                  0000  00000            .WORD   ^M<>
                                50          04    BC    D0 00002         MOVL    @4(R12),LO_LIM
                                5C          08    BC    D0 00006         MOVL    @8(R12),HI_LIM
                51 00000010G    EF 00000014G      EF    C3 0000A         SUBL3   IDATA+20,IDATA+16,R1               ; 1592
                                05                51    D1 00016         CMPL    R1,#5
                                                  00V   18 00019         BGEQ    8$
                                50 00000014G      EF    D1 0001B 2$:     CMPL    IDATA+20,LO_LIM                  ; 1598
                                                  00V   15 00022         BLEQ    4$
```

```
                                    000000014G  EF    D7 00024          DECL    IDATA+20                                ; 1600
                                 5C 00000010G  EF    D1 0002A 4$:      CMPL    IDATA+16,HI_LIM                         ; 1602
                                               00V   18 00031          BGEQ    6$
                                    00000010G  EF    D6 00033          INCL    IDATA+16                                ; 1604
                              51 00000010G  EF 00000014G  EF    C3 00039 6$:      SUBL3   IDATA+20,IDATA+16,R1
                                 04                       51    D1 00045          CMPL    R1,#4
                                                          D1    15 00048          BLEQ    2$
                                                                04 0004A 8$:      RET                                     ; 1610

; Routine Size: 75 bytes,    Routine Base: $CODE + 07FC1

                                                                00000 AUTO_SCALE:                                         ; 1656
                                                             0004 00000          .WORD   ^M<R2>
                                    50       04    BC          D0 00002          MOVL    @4(R12),LOW_LIMIT
                                    5C       08    BC          D0 00006          MOVL    @8(R12),HIGH_LIMIT
                                 00000000G  EF 00000010G  EF    D0 0000A          MOVL    IDATA+16,TEMP_INT2              ; 1664
                              51 00000010G  EF 00000014G  EF    C3 00015          SUBL3   IDATA+20,IDATA+16,R1            ; 1665
                                 51                       51    4E 00021          CVTLF   R1,R1
                        00000000G  EF    51                24    47 00024          DIVF3   #^F12.0,R1,TEMP_REAL
                                 00000018G  EF 00000000G  EF    4A 0002C          CVTFL   TEMP_REAL,IDATA+24              ; 1667
                              51 00000010G  EF 00000014G  EF    C3 00037          SUBL3   IDATA+20,IDATA+16,R1            ; 1669
                     51          51          00             00    7A 00043          EMUL    #0,#0,R1,R1
                     51          51          51             0C    7B 00048          EDIV    #12,R1,R1,R1
                                                          51    D5 0004D          TSTL    R1
                                                          00V   18 0004F          BGEQ    1$
                                    51                    0C    C0 00051          ADDL2   #12,R1
                                                          51    D5 00054 1$:      TSTL    R1
                                                          00V   15 00056          BLEQ    4$
                                    00000018G  EF         D6 00058          INCL    IDATA+24                                ; 1671
                           51 00000018G  EF               0C    C5 0005E 4$:      MULL3   #12,IDATA+24,R1                 ; 1678
                  00000010G  EF 00000014G  EF             51    C1 00066          ADDL3   R1,IDATA+20,IDATA+16
                                                          00V   11 00072          BRB     6$                              ; 1684
                                    00000014G  EF         D7 00074 5$:      DECL    IDATA+20                                ; 1688
                                    00000010G  EF         D7 0007A          DECL    IDATA+16                                ; 1689
                                 5C 00000010G  EF         D1 00080 6$:      CMPL    IDATA+16,HIGH_LIMIT
                                                          EB    14 00087          BGTR    5$
                                                          00V   11 00089          BRB     9$                              ; 1696
                                    00000014G  EF         D6 0008B 8$:      INCL    IDATA+20                                ; 1700
                                    00000010G  EF         D6 00091          INCL    IDATA+16                                ; 1701
                                 50 00000014G  EF         D1 00097 9$:      CMPL    IDATA+20,LOW_LIMIT
                                                          EB    19 0009E          BLSS    8$
                                                          00V   19 000A0          BLSS    12$                             ; 1708
                                 5C 00000010G  EF         D1 000A2          CMPL    IDATA+16,HIGH_LIMIT
                                                          00V   15 000A9          BLEQ    13$
                                    00000018G  EF         D7 000AB 12$:     DECL    IDATA+24                                ; 1710
                                 50 00000014G  EF         D1 000B1 13$:     CMPL    IDATA+20,LOW_LIMIT
                                                          A4    19 000B8          BLSS    4$
                                 5C 00000010G  EF         D1 000BA          CMPL    IDATA+16,HIGH_LIMIT
                                                          9B    14 000C1          BGTR    4$
                              00000000G  EF 00000010G  EF    D1 000C3          CMPL    IDATA+16,TEMP_INT2              ; 1717
                                                          00V   18 000CE          BGEQ    17$
                           50 00000000G  EF 00000010G  EF    C3 000D0          SUBL3   IDATA+16,TEMP_INT2,R0           ; 1721
                              00000014G  EF               50    C0 000DC          ADDL2   R0,IDATA+20
                           50 00000000G  EF 00000010G  EF    C3 000E3          SUBL3   IDATA+16,TEMP_INT2,R0           ; 1723
                              00000010G  EF               50    C0 000EF          ADDL2   R0,IDATA+16
                                                                04 000F6 17$:     RET                                     ; 1728
```

; Routine Size: 247 bytes, Routine Base: $CODE + 0800

```
                                    00000 ALT_SOURCE:                                                   ; 1780
                                00FC 00000          .WORD   ^M<R2,R3,R4,R5,R6,R7>
                          5E      14  C2 00002       SUBL2   #20,SP
                          52    04    BC 90 00005     MOVB    @4(R12),OBJ_TYP
                          53    08    BC 90 00009     MOVB    @8(R12),PRIM
                          54    0C    BC D0 0000D     MOVL    @12(R12),PRIMNUM
                          55    10    BC 90 00011     MOVB    @16(R12),SECO
                          56    14    BC D0 00015     MOVL    @20(R12),SECONUM
                          5C    18    BC 90 00019     MOVB    @24(R12),AN_FLAG
                          2E 00000000G  EF D1 0001D     CMPL    QTAB_OFFSET,#46                         ; 1794
                             00V        12 00024       BNEQ    2$
                             00000000G  EF 94 00026     CLRB    GLOBAL_SET                              ; 1796
                00V000000000G EF        00 E0 0002C 2$: BBS     #0,VISIBLE_QUESTION,4$                  ; 1803
                00V000000000G EF        00 E0 00034     BBS     #0,OPTIMIZING,6$
                   00000100   8F 00000000G  EF D1 0003C 4$: CMPL    QTAB_OFFSET,#256
                             00V        1E 00047       BGEQU   5$
                00VFFFFA850   EF 00000000G  EF E0 00049     BBS     QTAB_OFFSET,C.ALV,6$
                          57        01    90 00055 5$:    MOVB    #1,ALT_SOURCE                           ; 1812
                                0000V   31 00058        BRW     32$
                03 00000000G  EF        00 E0 0005B 6$: BBS     #0,OPTIMIZING,.+3                       ; 1816
                                0000V   31 00063        BRW     32$
                             00V        5C E9 00066     BLBC    AN_FLAG,9$                              ; 1823
                   00000000G  EF        00 FB 00069     CALLS   #0,POINT_AT_ANALYSIS                    ; 1825
                         FC   AD        56 D0 00070 9$: MOVL    SECONUM,-4(FP)                          ; 1827
                                   FC   AD 9F 00074     PUSHAB  -4(FP)
                         F8    AD        55 90 00077     MOVB    SECO,-8(FP)
                                   F8   AD 9F 0007B     PUSHAB  -8(FP)
                         F4    AD        54 D0 0007E     MOVL    PRIMNUM,-12(FP)
                                   F4   AD 9F 00082     PUSHAB  -12(FP)
                         F0    AD        53 90 00085     MOVB    PRIM,-16(FP)
                                   F0   AD 9F 00089     PUSHAB  -16(FP)
                         EC    AD        52 90 0008C     MOVB    OBJ_TYP,-20(FP)
                                   EC   AD 9F 00090     PUSHAB  -20(FP)
                   00000000G  EF        05 FB 00093     CALLS   #5,FIND_OBJECT
                             03         50 E8 0009A     BLBS    R0,.+3
                                0000V   31 0009D        BRW     24$
                                   57   94 000A0        CLRB    ALT_SOURCE                              ; 1831
                35            0B 00000000G  EF CF 000A2     CASEL   QTAB_OFFSET,#11,#53                 ; 1833
                                0000V      000AA        .DISPL  11$
                                0000V      000AC        .DISPL  11$
                                0000V      000AE        .DISPL  11$
                                006C       000B0        .DISPL  108
                                006C       000B2        .DISPL  108
                                0000V      000B4        .DISPL  13$
                                0000V      000B6        .DISPL  13$
                                0000V      000B8        .DISPL  13$
                                0000V      000BA        .DISPL  18$
                                0000V      000BC        .DISPL  18$
                                0000V      000BE        .DISPL  18$
                                006C       000C0        .DISPL  108
                                006C       000C2        .DISPL  108
                                006C       000C4        .DISPL  108
                                006C       000C6        .DISPL  108
                                006C       000C8        .DISPL  108
                                006C       000CA        .DISPL  108
```

```
                        0000V  000CC          .DISPL  18$
                        006C   000CE          .DISPL  108
                        0000V  000D0          .DISPL  18$
                        006C   000D2          .DISPL  108
                        0000V  000D4          .DISPL  13$
                        006C   000D6          .DISPL  108
                        006C   000D8          .DISPL  108
                        006C   000DA          .DISPL  108
                        006C   000DC          .DISPL  108
                        006C   000DE          .DISPL  10'
                        006C   000E0          .DISP'
                        0000V  000E2          .DISPL
                        0000V  000E4          .DISPL  13$
                        006C   000E6          .DISPL  108
                        006C   000E8          .DISPL  108
                        0000V  000EA          .DISPL  13$
                        006C   000EC          .DISPL  108
                        006C   000EE          .DISPL  108
                        0000V  000F0          .DISPL  17$
                        006C   000F2          .DISPL  108
                        006C   000F4          .DISPL  108
                        006C   000F6          .DISPL  108
                        006C   000F8          .DISPL  108
                        0000V  000FA          .DISPL  15$
                        006C   000FC          .DISPL  108
                        006C   000FE          .DISPL  108
                        0000V  00100          .DISPL  16$
                        0000V  00102          .DISPL  14$
                        006C   00104          .DISPL  108
                        0000V  00106          .DISPL  13$
                        0000V  00108          .DISPL  13$
                        0000V  0010A          .DISPL  13$
                        006C   0010C          .DISPL  108
                        006C   0010E          .DISPL  108
                        0000V  00110          .DISPL  13$
                        006C   00112          .DISPL  108
                        0000V  00114          .DISPL  14$
                        0000V  31 00116       BRW     22$
            50 00000000G   EF   D0 00119 11$: MOVL    QTAB_OFFSET,R0                  ; 1841
               FFFFFFB0GEF40   7F 00120       PUSHAQ     ,TA-80[R0]
            50 00000000G   EF   D0 00127       MOVL    DEF_CURRENT,R0
                     11    A0   9F 0012E       PUSHAB  17(R0)
00000000G   EF             02   FB 00131       CALLS   #2,LIB$SCOPY_DXDX
            50 00000000G   EF   D0 00138       MOVL    QTAB_OFFSET,R0                  ; 1842
FFFFFFF9GEF40              01   90 0013F       MOVB    #1,BDATA-7[R0]
                        0000V  31 00147       BRW     30$
            50 00000000G   EF   D0 0014A 13$:  MOVL    QTAB_OFFSET,R0                  ; 1859
            52 00000000G   EF   D0 00151       MOVL    DEF_CURRENT,R2
00000000GEF40        27    A2   D0 00158       MOVL    39(R2),IDATA[R0]
            50 00000000G   EF   D0 00161       MOVL    QTAB_OFFSET,R0                  ; 1860
00000000G   EF 00000000GEF40   D0 00168       MOVL    IDATA[R0],INPUT_VALUE
                        0000V  31 00174       BRW     30$
            50 00000000G   EF   D0 00177 14$:  MOVL    QTAB_OFFSET,R0                  ; 1870
            52 00000000G   EF   D0 0017E       MOVL    DEF_CURRENT,R2
00000000GEF40        23    A2   D0 00185       MOVL    35(R2),IDATA[R0]
            50 00000000G   EF   D0 0018E       MOVL    QTAB_OFFSET,R0                  ; 1871
00000000G   EF 00000000GEF40   D0 00195       MOVL    IDATA[R0],INPUT_VALUE
```

```
                                0000V  31 001A1          BRW     30$
                    50 00000000G   EF  D0 001A4 15$:     MOVL    QTAB_OFFSET,R0                  ; 1879
                    52 00000000G   EF  D0 001AB          MOVL    DEF_CURRENT,R2
          00000000GEF40        27    A2  D0 001B2          MOVL    39(R2),IDATA[R0]
                    50 00000000G   EF  D0 001BB          MOVL    QTAB_OFFSET,R0                  ; 1880
          00000000G  EF 00000000GEF40  D0 001C2          MOVL    IDATA[R0],INPUT_VALUE
                    50 00000000G   EF  D0 001CE          MOVL    SEGMENT_NUMBER,R0               ; 1881
          00000000GEF40 00000000G   EF  D0 001D5          MOVL    INPUT_VALUE,SEGMENT_POSITION[R0]
                                0000V  31 001E1          BRW     30$
                    50 00000000G   EF  D0 001E4 16$:     MOVL    QTAB_OFFSET,R0                  ; 1889
                    52 00000000G   EF  D0 001EB          MOVL    DEF_CURRENT,R2
          00000000GEF40        27    A2  D0 001F2          MOVL    39(R2),IDATA[R0]
                    50 00000000G   EF  D0 001FB          MOVL    QTAB_OFFSET,R0                  ; 1890
          00000000G  EF 00000000GEF40  D0 00202          MOVL    IDATA[R0],INPUT_VALUE
                    50 00000000G   EF  D0 0020E          MOVL    SEGMENT_NUMBER,R0               ; 1891
          00000000GEF40 00000000G   EF  D0 00215          MOVL    INPUT_VALUE,SEGMENT_LENGTH[R0]
                                0000V  31 00221          BRW     30$
                    50 00000000G   EF  D0 00224 17$:     MOVL    QTAB_OFFSET,R0                  ; 1899
                    52 00000000G   EF  D0 0022B          MOVL    DEF_CURRENT,R2
          00000000GEF40        27    A2  D0 00232          MOVL    39(R2),IDATA[R0]
                    50 00000000G   EF  D0 0023B          MOVL    QTAB_OFFSET,R0                  ; 1900
          00000000G  EF 00000000GEF40  D0 00242          MOVL    IDATA[R0],INPUT_VALUE
          00000000G  EF                01  90 0024E          MOVB    #1,GLOBAL_SET                   ; 1901
                                0000V  31 00255          BRW     30$
                    50 00000000G   EF  D0 00258 18$:     MOVL    QTAB_OFFSET,R0                  ; 1913
                    52 00000000G   EF  D0 0025F          MOVL    DEF_CURRENT,R2
          FFFFFFF9GEF40        2B    A2  90 00266          MOVB    43(R2),BDATA-7[R0]
                    50 00000000G   EF  D0 0026F          MOVL    DEF_CURRENT,R0                  ; 1915
       00V      2B   A0                00  E1 00276          BBC     #0,43(R0),20$
          00000000G  EF                01  D0 0027B          MOVL    #1,INPUT_VALUE                  ; 1917
                                00V    11  00282          BRB     30$
                       00000000G   EF  D4 00284 20$:     CLRL    INPUT_VALUE                     ; 1921
                                00V    11  0028A          BRB     30$
                                00V    11  0028C 22$:     BRB     30$
                 57                    01  90 0028E 24$:     MOVB    #1,ALT_SOURCE                   ; 1940
       00V00000000G  EF                00  E1 00291          BBC     #0,AUTO_TUNE,27$                ; 1946
          00000100   8F 00000000G   EF  D1 00299          CMPL    QTAB_OFFSET,#256
                                00V    1E  002A4          BGEQU   27$
       00VFFFFA613   EF 00000000G   EF  E1 002A6          BBC     QTAB_OFFSET,C.ALW,27$
                                      00  DD 002B2          PUSHL   #0                              ; 1953
                                      00  DD 002B4          PUSHL   #0
                                      00  DD 002B6          PUSHL   #0
                       00B3801C    8F    DD 002B8          PUSHL   #11763740
          00000000G  EF                04  FB 002BE          CALLS   #4,LIB$STOP
          00000100   8F 00000000G   EF  D1 002C5 27$:     CMPL    QTAB_OFFSET,#256                ; 1955
                                00V    1E  002D0          BGEQU   30$
       00VFFFFA607   EF 00000000G   EF  E1 002D2          BBC     QTAB_OFFSET,C.ALX,30$
                                57    94  002DE          CLRB    ALT_SOURCE                      ; 1960
                    50 00000000G   EF  D0 002E0          MOVL    QTAB_OFFSET,R0                  ; 1961
                       FFFFFFE0GEF40    D4 002E7          CLRF    RDATA-32[R0]
          00000000G  EF                00  FB 002EE 30$:     CALLS   #0,POINT_AT_DEFINITION          ; 1967
                    50                 57  90 002F5 32$:     MOVB    ALT_SOURCE,R0                   ; 1971
                                           04 002F8          RET

; Routine Size: 761 bytes,    Routine Base: $CODE + 08103

                                      00000 PRE_PROCESS:                                           ; 2016
```

```
                                 007C 00000          .WORD   ^M<R2,R3,R4,R5,R6>
                        5C    01   90 00002          MOVB    #1,PRE_PROCESS                          ; 2028
                     3B    0B 00000000G  EF   CF 00005          CASEL   QTAB_OFFSET,#11,#59             ; 2030
                                    0000V      0000D          .DISPL  129$
                                    0000V      0000F          .DISPL  132$
                                    0000V      00011          .DISPL  135$
                                    0000V      00013          .DISPL  138$
                                    0000V      00015          .DISPL  138$
                                    0000V      00017          .DISPL  17$
                                    0000V      00019          .DISPL  21$
                                    0000V      0001B          .DISPL  25$
                                    0000V      0001D          .DISPL  64$
                                    0000V      0001F          .DISPL  69$
                                    0000V      00021          .DISPL  75$
                                    0000V      00023          .DISPL  141$
                                    0000V      00025          .DISPL  103$
                                    0000V      00027          .DISPL  125$
                                    0078       00029          .DISPL  120
                                    0000V      0002B          .DISPL  81$
                                    0000V      0002D          .DISPL  85$
                                    0000V      0002F          .DISPL  47$
                                    0000V      00031          .DISPL  39$
                                    0000V      00033          .DISPL  51$
                                    0078       00035          .DISPL  120
                                    0000V      00037          .DISPL  56$
                                    0000V      00039          .DISPL  183$
                                    0000V      0003B          .DISPL  110$
                                    0078       0003D          .DISPL  120
                                    0000V      0003F          .DISPL  8$
                                    0000V      00041          .DISPL  147$
                                    0078       00043          .DISPL  120
                                    0000V      00045          .DISPL  186$
                                    0000V      00047          .DISPL  191$
                                    0078       00049          .DISPL  120
                                    0000V      0004B          .DISPL  229$
                                    0000V      0004D          .DISPL  196$
                                    0078       0004F          .DISPL  120
                                    0078       00051          .DISPL  120
                                    0000V      00053          .DISPL  91$
                                    0078       00055          .DISPL  120
                                    0000V      00057          .DISPL  92$
                                    0078       00059          .DISPL  120
                                    0000V      0005B          .DISPL  8$
                                    0000V      0005D          .DISPL  29$
                                    0000V      0005F          .DISPL  12$
                                    0078       00061          .DISPL  120
                                    0000V      00063          .DISPL  163$
                                    0000V      00065          .DISPL  195$
                                    0000V      00067          .DISPL  115$
                                    0000V      00069          .DISPL  200$
                                    0000V      0006B          .DISPL  205$
                                    0000V      0006D          .DISPL  80$
                                    0000V      0006F          .DISPL  55$
                                    0078       00071          .DISPL  120
                                    0000V      00073          .DISPL  178$
                                    0078       00075          .DISPL  120
                                    0000V      00077          .DISPL  223$
```

```
                                                0078      00079          .DISPL  120
                                                0078      0007B          .DISPL  120
                                                0078      0007D          .DISPL  120
                                                0000V     0007F          .DISPL  204$
                                                0078      00081          .DISPL  120
                                                0000V     00083          .DISPL  1$
                                                0000V  31 00085          BRW     288$
                                                  50   94 00088 1$:      CLRB    R0                                      ; 2034
                      51 00000000G   EF           19   C5 0008A          MULL3   #25,QTAB_OFFSET,R1
                                     51 FFFFFEF2GEF41   9E 00092          MOVAB   QTAB-270[R1],R1
                         00000100    8F             61   D1 0009A          CMPL    (R1),#256
                                                   00V  1E 000A1          BGEQU   3$
                      00VFFFFA561    EF             61   E1 000A3          BBC     (R1),C.ALY,3$
                                                   50   96 000AB          INCB    R0
                                                   52   94 000AD 3$:      CLRB    R2
                                        00000084G  EF   D5 000AF          TSTL    IDATA+132
                                                   00V  13 000B5          BEQL    5$
                                                   52   96 000B7          INCB    R2
                                     52            50   8A 000B9 5$:      BICB2   R0,R2
                                     00V           52   E9 000BC          BLBC    R2,7$
                                     61            05   D0 000BF          MOVL    #5,(R1)                                 ; 2041
                                                0000V  31 000C2 7$:      BRW     289$
                                        00000014G  EF   D5 000C5 8$:      TSTL    IDATA+20                                ; 2046
                                                   00V  12 000CB          BNEQ    10$
                         00000000G  EF 000186A0    8F   D0 000CD          MOVL    #100000,DEF                             ; 2048
                                                0000V  31 000D8          BRW     289$
               00000000G EF 00000014G  EF           32   C5 000DB 10$:     MULL3   #50,IDATA+20,DEF                        ; 2052
                                                0000V  31 000E7          BRW     289$
                                        000000E4G  EF   D5 000EA 12$:     TSTL    IDATA+228                               ; 2061
                                                   00V  13 000F0          BEQL    14$
000000E4G  EF        FF    8F             08           00   ED 000F2          CMPZV   #0,#8,#^XFF,IDATA+228
                                                   00V  18 000FC          BGEQ    15$
                         00000000G  EF        FF   8F   9A 000FE 14$:     MOVZBL  #255,MAX_KEY_SIZE                       ; 2067
                                                   00V  11 00106          BRB     16$
                         00000000G  EF 000000E4G  EF   D0 00108 15$:     MOVL    IDATA+228,MAX_KEY_SIZE                  ; 2071
                      51 00000000G  EF             19   C5 00113 16$:     MULL3   #25,QTAB_OFFSET,R1                      ; 2073
               FFFFFEFAGEF41 00000000G  EF           04   28 0011B          MOVC3   #4,MAX_KEY_SIZE,QTAB-262[R1]
                      50 00000000G  EF             19   C5 00128          MULL3   #25,QTAB_OFFSET,R0                      ; 2074
               FFFFFF13GEF40 00000000G  EF           04   28 00130          MOVC3   #4,MAX_KEY_SIZE,QTAB-237[R0]
                      50 00000000G  EF             19   C5 0013D          MULL3   #25,QTAB_OFFSET,R0                      ; 2075
               FFFFFF0BGEF40 00000000G  EF           04   28 00145          MOVC3   #4,MAX_KEY_SIZE,QTAB-245[R0]
                                                0000V  31 00152          BRW     289$
                                                   5C   94 00155 17$:     CLRB    PRE_PROCESS                             ; 2083
                                     50 00000000G  EF   D0 00157          MOVL    QTAB_OFFSET,R0                          ; 2084
                                        00000000GEF40   D4 0015E          CLRL    IDATA[R0]
                      03 00000033G   EF             00   E0 00165          BBS     #0,VDATA+51,.+3                         ; 2086
                                                0000V  31 0016D          BRW     289$
                                     02 000000F8G  EF   D1 00170          CMPL    IDATA+248,#2
                                                   03   14 00177          BGTR    .+3
                                                0000V  31 00179          BRW     289$
                                              01   8F   9F 0017C          PUSHAB  #1                                      ; 2092
                                        00000000   8F   DF 0017F          PUSHAL  #0
                                              0B   8F   9F 00185          PUSHAB  #11
                                        00000084G  EF   9F 00188          PUSHAB  IDATA+132
                                              04   8F   9F 0018E          PUSHAB  #4
                                              01   8F   9F 00191          PUSHAB  #1
                            8103   CF             06   FB 00194          CALLS   #6,ALT_SOURCE
```

```
                    56              50   90 00199          MOVB    R0,RESULT
                                0000V    31 0019C          BRW     289$
                                   5C    94 0019F 21$:     CLRB    PRE_PROCESS                                 ; 2101
                    50 00000000G   EF    D0 001A1          MOVL    QTAB_OFFSET,R0                              ; 2102
                       00000000GEF 40    D4 001A8          CLRL    IDATA[R0]
          03 00000033G  EF              00   E0 001AF          BBS     #0,VDATA+51,.+3                             ; 2104
                                0000V    31 001B7          BRW     289$
                    02 000000F8G  EF    D1 001BA          CMPL    IDATA+248,#2
                                   03    14 001C1          BGTR    .+3
                                0000V    31 001C3          BRW     289$
                               01   8F   9F 001C6          PUSHAB  #1                                          ; 2110
                          00000000   8F   DF 001C9          PUSHAL  #0
                               0C   8F   9F 001CF          PUSHAB  #12
                          00000084G  EF   9F 001D2          PUSHAB  IDATA+132
                               04   8F   9F 001D8          PUSHAB  #4
                               01   8F   9F 001DB          PUSHAB  #1
                    8103   CF              06   FB 001DE          CALLS   #6,ALT_SOURCE
                    56              50   90 001E3          MOVB    R0,RESULT
                                0000V    31 001E6          BRW     289$
                                   5C    94 001E9 25$:     CLRB    PRE_PROCESS                                 ; 2119
                    50 00000000G   EF    D0 001EB          MOVL    QTAB_OFFSET,R0                              ; 2120
                       00000000GEF 40    D4 001F2          CLRL    IDATA[R0]
          03 00000033G  EF              00   E0 001F9          BBS     #0,VDATA+51,.+3                             ; 2122
                                0000V    31 00201          BRW     289$
                    02 000000F8G  EF    D1 00204          CMPL    IDATA+248,#2
                                   03    14 0020B          BGTR    .+3
                                0000V    31 0020D          BRW     289$
                               01   8F   9F 00210          PUSHAB  #1                                          ; 2128
                          00000000   8F   DF 00213          PUSHAL  #0
                               12   8F   9F 00219          PUSHAB  #18
                          00000084G  EF   9F 0021C          PUSHAB  IDATA+132
                               04   8F   9F 00222          PUSHAB  #4
                               01   8F   9F 00225          PUSHAB  #1
                    8103   CF              06   FB 00228          CALLS   #6,ALT_SOURCE
                    56              50   90 0022D          MOVB    R0,RESULT
                                0000V    31 00230          BRW     289$
          00V00000013G  EF              00   E1 00233 29$:     BBC     #0,BDATA+19,34$                             ; 2140
                                   50    D4 0023B          CLRL    TEMP_KEY_SIZE                               ; 2144
                                   51    D4 0023D          CLRL    R1                                          ; 2146
                       00000000G  EF    51   D0 0023F 31$:     MOVL    R1,TEMP_INT2
                                   52 00000000G   EF    D0 00246          MOVL    TEMP_INT2,R2                                ; 2150
          00V00000000GEF 42              00   E1 0024D          BBC     #0,SEGMENT_WANTED[R2],33$
                                   52 00000000G   EF    D0 00256          MOVL    TEMP_INT2,R2                                ; 2152
                                   50 00000000GEF 42    C0 0025D          ADDL2   SEGMENT_LENGTH[R2],TEMP_KEY_SIZE
          D6                       51              07   F3 00265 33$:     AOBLEQ  #7,R1,3T$
                                                   00V  11 00269          BRB     35$
                                   50 000000D8G   EF    D0 0026B 34$:     MOVL    IDATA+216,TEMP_KEY_SIZE                     ; 2161
                                      000000E4G   EF    D5 00272 35$:     TSTL    IDATA+228                                   ; 2163
                                                   00V  12 00278          BNEQ    37$
00000000G  EF 00000000G   EF                       50   C3 0027A          SUBL3   TEMP_KEY_SIZE,CUR_MAX_REC,MAX_KEY_POSITION  ; 2165
                                                   00V  11 00286          BRB     38$
00000000G  EF 000000E4G   EF                       50   C3 00288 37$:     SUBL3   TEMP_KEY_SIZE,IDATA+228,MAX_KEY_POSITION    ; 2169
           50 00000000G   EF                       19   C5 00294 38$:     MULL3   #25,QTAB_OFFSET,R0                          ; 2173
FFFFFEFAGEF40 00000000G   EF                       04   28 0029C          MOVC3   #4,MAX_KEY_POSITION,QTAB-262[R0]
                                                  00   8F   9F 002A9          PUSHAB  #0                                          ; 2175
                                      00000000G   EF    9F 002AC          PUSHAB  SEGMENT_NUMBER
                                             86   8F   9F 002B2          PUSHAB  #-122
```

```
                    00000084G  EF   9F 002B5          PUSHAB  IDATA+132
                          0B   8F   9F 002BB          PUSHAB  #11
                          01   8F   9F 002BE          PUSHAB  #1
                8103      CF         06 FB 002C1      CALLS   #6,ALT_SOURCE
                          5C         50 90 002C6      MOVB    R0,PRE_PROCESS
                                0000V 31 002C9      BRW     289$
                    00000084G  EF   D5 002CC 39$:    TSTL    IDATA+132                                   ; 2182
                                00V 13 002D2        BEQL    41$
          00V00000012G  EF          00 E0 002D4      BBS     #0,VDATA+18,45$
                    00000088G  EF   D5 002DC 41$:    TSTL    IDATA+136                                   ; 2188
                                00V 15 002E2        BLEQ    43$
                          5C         01 90 002E4      MOVB    #1,PRE_PROCESS                              ; 2190
                                0000V 31 002E7      BRW     289$
                       50 00000000G  EF   D0 002EA 43$:    MOVL    QTAB_OFFSET,R0                      ; 2196
                FFFFFFF9GEF40         01 90 002F1      MOVB    #1,BDATA-7[R0]
                                      5C 94 002F9      CLRB    PRE_PROCESS                                 ; 2197
                                0000V 31 002FB      BRW     289$
                                      5C 94 002FE 45$:    CLRB    PRE_PROCESS                             ; 2203
                                0000V 31 00300      BRW     289$
                    00000084G  EF   D5 00303 47$:    TSTL    IDATA+132                                   ; 2207
                                00V 13 00309        BEQL    49$
                          00   8F   9F 0030B        PUSHAB  #0                                          ; 2209
                    00000000   8F   DF 0030E        PUSHAL  #0
                          77   8F   9F 00314        PUSHAB  #119
                    00000084G  EF   9F 00317        PUSHAB  IDATA+132
                          0B   8F   9F 0031D        PUSHAB  #11
                          01   8F   9F 00320        PUSHAB  #1
                8103      CF         06 FB 00323      CALLS   #6,ALT_SOURCE
                          5C         50 90 00328      MOVB    R0,PRE_PROCESS
                                0000V 31 0032B      BRW     289$
                                      5C 94 0032E 49$:    CLRB    PRE_PROCESS                             ; 2216
                       50 00000000G  EF   D0 00330      MOVL    QTAB_OFFSET,R0                          ; 2217
                          FFFFFFF9GEF40   94 00337      CLRB    BDATA-7[R0]
                                0000V 31 0033E      BRW     289$
          50 00000000G  EF          19 C5 00341 51$:    MULL3   #25,QTAB_OFFSET,R0                      ; 2225
                       50 FFFFFEF2GEF40   9E 00349      MOVAB   QTAB-270[R0],R0
                          00000084G  EF   D5 00351      TSTL    IDATA+132
                                00V 12 00357        BNEQ    53$
                                60 D4 00359         CLRL    (R0)                                        ; 2227
                                00V 11 0035B        BRB     54$
                       60               01 D0 0035D 53$:    MOVL    #1,(R0)                             ; 2231
                                00   8F   9F 00360 54$:    PUSHAB  #0                                  ; 2233
                          00000000   8F   DF 00363        PUSHAL  #0
                                7C   8F   9F 00369        PUSHAB  #124
                          00000084G  EF   9F 0036C        PUSHAB  IDATA+132
                                0B   8F   9F 00372        PUSHAB  #11
                                01   8F   9F 00375        PUSHAB  #1
                      8103      CF         06 FB 00378      CALLS   #6,ALT_SOURCE
                                5C         50 90 0037D      MOVB    R0,PRE_PROCESS
                                      0000V 31 00380      BRW     289$
                       50 00000000G  EF   92 00383 55$:    MCOMB   NUMBER_KEYS_SET,R0                  ; 2240
          5C 00000000G  EF          50 89 0038A      BISB3   R0,VISIBLE_QUESTION,PRE_PROCESS
                                0000V 31 00392      BRW     289$
          00V00000000G  EF          00 E1 00395 56$:    BBC     #0,OPTIMIZING,58$                       ; 2254
                                01   8F   9F 0039D        PUSHAB  #1                                  ; 2256
                          00000000   8F   DF 003A0        PUSHAL  #0
                                4B   8F   9F 003A6        PUSHAB  #75
```

```
                    00000000   8F   DF 003A9          PUSHAL  #0
                          08   8F   9F 003AF          PUSHAB  #8
                          01   8F   9F 003B2          PUSHAB  #1
                8103      CF             06   FB 003B5          CALLS   #6,ALT_SOURCE
                          5C             50   90 003BA          MOVB    R0,PRE_PROCESS
                                         00V  11 003BD          BRB     63$
                    00000084G   EF   D5 003BF 58$:    TSTL    IDATA+132                       ; 2260
                                         00V  13 003C5          BEQL    60$
          00V00000017G      EF             00   E0 003C7          BBS     #0,VDATA+23,61$
                          5C             01   90 003CF 60$:    MOVB    #1,PRE_PROCESS
                                         00V  11 003D2          BRB     62$
                                         5C   94 003D4 61$:    CLRB    PRE_PROCESS
                                              003D6 62$:
                                       0000V  31 003D6 63$:    BRW     289$
                                         5C   94 003D9 64$:    CLRB    PRE_PROCESS             ; 2272
                          50 00000000G   EF   D0 003DB          MOVL    QTAB_OFFSET,R0          ; 2273
                             FFFFFFF9GEF40   94 003E2          CLRB    BDATA-7[R0]
          00V00000033G      EF             00   E1 003E9          BBC     #0,VDATA+51,68$         ; 2275
                          02 000000F8G   EF   D1 003F1          CMPL    IDATA+248,#2
                                         00V  15 003F8          BLEQ    68$
                          21 000000DCG   EF   D1 003FA          CMPL    IDATA+220,#33
                                         00V  12 00401          BNEQ    68$
                                00   8F   9F 00403          PUSHAB  #0                      ; 2283
                    00000000   8F   DF 00406          PUSHAL  #0
                          7A   8F   9F 0040C          PUSHAB  #122
                    00000084G   EF   9F 0040F          PUSHAB  IDATA+132
                          0B   8F   9F 00415          PUSHAB  #11
                          01   8F   9F 00418          PUSHAB  #1
                8103      CF             06   FB 0041B          CALLS   #6,ALT_SOURCE
                          5C             50   90 00420          MOVB    R0,PRE_PROCESS
                                       0000V  31 00423 68$:    BRW     289$
                                         5C   94 00426 69$:    CLRB    PRE_PROCESS             ; 2292
                          50 00000000G   EF   D0 00428          MOVL    QTAB_OFFSET,R0          ; 2293
                             FFFFFFF9GEF40   94 0042F          CLRB    BDATA-7[R0]
          03 00000033G      EF             00   E0 00436          BBS     #0,VDATA+51,.+3         ; 2295
                                       0000V  31 0043E          BRW     289$
                          02 000000F8G   EF   D1 00441          CMPL    IDATA+248,#2
                                         03   14 00448          BGTR    .+3
                                       0000V  31 0044A          BRW     289$
                          21 000000DCG   EF   D1 0044D          CMPL    IDATA+220,#33
                                         03   13 00454          BEQL    .+3
                                       0000V  31 00456          BRW     289$
                             000000084G   EF   D5 00459          TSTL    IDATA+132
                                         03   13 0045F          BEQL    .+3
                                       0000V  31 00461          BRW     289$
                                00   8F   9F 00464          PUSHAB  #0                      ; 2305
                    00000000   8F   DF 00467          PUSHAL  #0
                          7B   8F   9F 0046D          PUSHAB  #123
                    00000084G   EF   9F 00470          PUSHAB  IDATA+132
                          0B   8F   9F 00476          PUSHAB  #11
                          01   8F   9F 00479          PUSHAB  #1
                8103      CF             06   FB 0047C          CALLS   #6,ALT_SOURCE
                          5C             50   90 00481          MOVB    R0,PRE_PROCESS
                                       0000V  31 00484          BRW     289$
                                         5C   94 00487 75$:    CLRB    PRE_PROCESS             ; 2314
                          50 00000000G   EF   D0 00489          MOVL    QTAB_OFFSET,R0          ; 2315
                             FFFFFFF9GEF40   94 00490          CLRB    BDATA-7[R0]
```

```
                00V00000033G  EF                00  E1 00497          BBC     #0,VDATA+51,79$                          ; 2317
                              02 000000F8G      EF  D1 0049F          CMPL    IDATA+248,#2
                                                00V 15 004A6          BLEQ    79$
                              21 000000DCG      EF  D1 004A8          CMPL    IDATA+220,#33
                                                00V 12 004AF          BNEQ    79$
                                       00       8F  9F 004B1          PUSHAB  #0                                       ; 2325
                                 00000000       8F  DF 004B4          PUSHAL  #0
                                       7E       8F  9F 004BA          PUSHAB  #126
                                 00000084G      EF  9F 004BD          PUSHAB  IDATA+132
                                       0B       8F  9F 004C3          PUSHAB  #11
                                       01       8F  9F 004C6          PUSHAB  #1
                           8103  CF               06  FB 004C9          CALLS   #6,ALT_SOURCE
                                 5C               50  90 004CE          MOVB    R0,PRE_PROCESS
                                            0000V  31  004D1 79$:       BRW     289$
                                                5C  94  004D4 80$:      CLRB    PRE_PROCESS                              ; 2334
                                 000000ECG      EF  D4 004D6          CLRL    IDATA+236                                ; 2335
                                       01       8F  9F 004DC          PUSHAB  #1                                       ; 2337
                                 00000000       8F  DF 004DF          PUSHAL  #0
                                       11       8F  9F 004E5          PUSHAB  #17
                                 00000084G      EF  9F 004E8          PUSHAB  IDATA+132
                                       04       8F  9F 004EE          PUSHAB  #4
                                       01       8F  9F 004F1          PUSHAB  #1
                           8103  CF               06  FB 004F4          CALLS   #6,ALT_SOURCE
                                 56               50  90 004F9          MOVB    R0,RESULT
                                            0000V  31  004FC          BRW     289$
                                                50  D4  004FF 81$:      CLRL    R0                                       ; 2346
                     00000000G  EF                50  D0 00501 82$:     MOVL    R0,TEMP_INT2
                                 51 00000000G     EF  D0 00508          MOVL    TEMP_INT2,R1                             ; 2350
                                    00000000GEF   41  94 0050F          CLRB    SEGMENT_WANTED[R1]
                                 51 00000000G     EF  D0 00516          MOVL    TEMP_INT2,R1                             ; 2351
                                    00000000GEF   41  D4 0051D          CLRL    SEGMENT_POSITION[R1]
                                 51 00000000G     EF  D0 00524          MOVL    TEMP_INT2,R1                             ; 2352
                                    00000000GEF   41  D4 0052B          CLRL    SEGMENT_LENGTH[R1]
                           CB                    50      07  F3 00532    AOBLEQ  #7,R0,82$
                                 50 00000000G     EF  D0 00536          MOVL    QTAB_OFFSET,R0                           ; 2356
                                    FFFFFFF9GEF   40  94 0053D          CLRB    BDATA-7[R0]
                                 21 000000DCG     EF  D1 00544          CMPL    IDATA+220,#33                            ; 2358
                                                   03  12 0054B          BNEQ    .+3
                                               0000V  31  0054D          BRW     289$
                                                   5C  94  00550          CLRB    PRE_PROCESS                              ; 2360
                                               0000V  31  00552          BRW     289$
                00V00000000G  EF                   00  E1 00555 85$:     BBC     #0,OPTIMIZING,90$                        ; 2375
                                 00000000       8F  DF 0055D          PUSHAL  #0                                       ; 2379
                                       55       8F  9F 00563          PUSHAB  #85
                                 00000000       8F  DF 00566          PUSHAL  #0
                                       08       8F  9F 0056C          PUSHAB  #8
                                       01       8F  9F 0056F          PUSHAB  #1
                     00000000G  EF                   05  FB 00572          CALLS   #5,FIND_OBJECT
                                00V                 50  E9 00579          BLBC    R0,88$
                     00000000G  EF                   01  D0 0057C          MOVL    #1,INPUT_VALUE                   ; 2381
                                                   00V 11 00583          BRB     89$
                                 00000000G      EF  D4 00585 88$:       CLRL    INPUT_VALUE                              ; 2385
                                                5C  94 0058B 89$:       CLRB    PRE_PROCESS                              ; 2387
                                            0000V  31 0058D 90$:        BRW     289$
                                                5C  94 00590 91$:       CLRB    PRE_PROCESS                              ; 2397
                                       00       8F  9F 00592          PUSHAB  #0                                       ; 2398
                                 00000000       8F  DF 00595          PUSHAL  #0
```

```
                                55   8F   9F 0059B          PUSHAB  #85
                          00000000   8F   DF 0059E          PUSHAL  #0
                                08   8F   9F 005A4          PUSHAB  #8
                                01   8F   9F 005A7          PUSHAB  #1
                      8103      CF        06   FB 005AA    CALLS   #6,ALT_SOURCE
                            56          50   90 005AF    MOVB    R0,RESULT
                                  0000V 31   005B2    BRW     289$
                          00000084G     EF   D5 005B5 92$:  TSTL    IDATA+132                       ; 2410
                                  00V   13   005BB    BEQL    94$
                            50 00090000G  EF   D0 005BD    MOVL    QTAB_OFFSET,R0
                  00VFFFFFFF5GEF40        00   E0 005C4    BBS     #0,VDATA-11[R0],95$
                            5C            01   90 005CD 94$:  MOVB    #1,PRE_PROCESS
                                  00V     11   005D0    BRB     96$
                                  5C      94   005D2 95$:  CLRB    PRE_PROCESS
                  00V00000000G    EF            00   E1 005D4 96$:  BBC     #0,OPTIMIZING,101$          ; 2416
                     00000000G    EF            00   FB 005DC    CALLS   #0,POINT_AT_ANALYSIS        ; 2420
                          00000000   8F   DF 005E3          PUSHAL  #0                              ; 2422
                                0D   8F   9F 005E9          PUSHAB  #13
                          00000000   8F   DF 005EC          PUSHAL  #0
                                04   8F   9F 005F2          PUSHAB  #4
                                01   8F   9F 005F5          PUSHAB  #1
                        00000000G    EF            05   FB 005F8    CALLS   #5,FIND_OBJECT
                                  00V           50   E9 005FF    BLBC    R0,99$
                                  50 00000000G  EF   D0 00602    MOVL    DEF_CURRENT,R0              ; 2424
                        00000000G    EF      27    A0   D0 00609    MOVL    39(R0),OLD_COUNT
                                  00V           11   00611    BRB     100$
                          00000000G     EF   D4 00613 99$:  CLRL    OLD_COUNT                       ; 2428
                     00000000G    EF            00   FB 00619 100$:  CALLS   #0,POINT_AT_DEFINITION   ; 2430
                  50 00000000G    EF            19   C5 00620    MULL3   #25,QTAB_OFFSET,R0          ; 2432
                     FFFFFEF1GEF40              01   90 00628    MOVB    #1,QTAB-271[R0]
                  50 00000000G    EF            19   C5 00630    MULL3   #25,QTAB_OFFSET,R0          ; 2433
            FFFFFEF2GEF40 00000000G    EF            04   28 00638    MOVC3   #4,OLD_COUNT,QTAB-270[R0]
                                  00V           11   00645    BRB     102$
                  50 00000000G    EF            19   C5 00647 101$:  MULL3   #25,QTAB_OFFSET,R0          ; 2439
                     FFFFFEF1GEF40              94   0064F    CLRB    QTAB-271[R0]
                                0000V 31   00656 102$:  BRW     289$
                          000000C0G     EF   D5 00659 103$:  TSTL    IDATA+192                       ; 2449
                                  00V   15   0065F    BLEQ    108$
                          000000E0G     EF   D5 00661    TSTL    IDATA+224
                                  00V   13   00667    BEQL    108$
                          00000084G     EF   D5 00669    TSTL    IDATA+132
                                  00V   13   0066F    BEQL    107$
                  00V0000000CG    EF            00   E0 00671    BBS     #0,VDATA+12,108$
                                  5C            01   90 00679 107$:  MOVB    #1,PRE_PROCESS
                                  00V           11   0067C    BRB     109$
                                  5C      94   0067E 108$:  CLRB    PRE_PROCESS
                                0000V 31   00680 109$:  BRW     289$
                          00000084G     EF   D5 00683 110$:  TSTL    IDATA+132                       ; 2467
                                  00V   13   00689    BEQL    112$
                  00V000000017G   EF            00   E0 0068B    BBS     #0,VDATA+23,113$
                                  5C            01   90 00693 112$:  MOVB    #1,PRE_PROCESS
                                0000V 31   00696    BRW     289$
                                  5C      94   00699 113$:  CLRB    PRE_PROCESS
                                0000V 31   0069B    BRW     289$
                          000000C0G     EF   D5 0069E 115$:  TSTL    IDATA+192                       ; 2477
                                  00V   15   006A4    BLEQ    121$
                          00000084G     EF   D5 006A6    TSTL    IDATA+132                       ; 2479
```

```
                    00V  13 006AC          BEQL    118$
                 50 00000000G   EF  D0 006AE          MOVL    QTAB_OFFSET,R0
00VFFFFFFF5GEF40                00  E0 006B5          BBS     #0,VDATA-11[R0],119$
                 56             01  90 006BE 118$:    MOVB    #1,RESULT
                                00V 11 006C1          BRB     122$
                                56  94 006C3 119$:    CLRB    RESULT
                                00V 11 006C5          BRB     122$
                 50 00000000G   EF  D0 006C7 121$:    MOVL    QTAB_OFFSET,R0                          ; 2492
   00000000GEF40                02  D0 006CE          MOVL    #2,IDATA[R0]
                                56  94 006D6          CLRB    RESULT                                  ; 2493
              00V               56  E8 006D8 122$:    BLBS    RESULT,124$                             ; 2497
                 50 00000000G   EF  D0 006DB          MOVL    QTAB_OFFSET,R0                          ; 2499
   00000000G  EF 00000000GEF40  D0 006E2          MOVL    IDATA[R0],INPUT_VALUE
              5C                56  90 006EE 124$:    MOVB    RESULT,PRE_PROCESS                      ; 2501
                           0000V  31 006F1          BRW     289$
              04 00000108G   EF  D1 006F4 125$:    CMPL    IDATA+264,#4                            ; 2509
                                00V 12 006FB          BNEQ    127$
              5C                01  90 006FD          MOVB    #1,PRE_PROCESS                          ; 2511
                           0000V  31 00700          BRW     289$
                                5C  94 00703 127$:    CLRB    PRE_PROCESS                             ; 2517
   00000000G  EF                01  D0 00705          MOVL    #1,INPUT_VALUE                          ; 2518
                           0000V  31 0070C          BRW     289$
   00000000G  EF          7E    8F  9A 0070F 129$:    MOVZBL  #126,MAX_STRING_ANSWER_LENGTH           ; 2528
              50 00000000G   EF  D0 00717          MOVL    QTAB_OFFSET,R0                          ; 2529
                 FFFFFFF9GEF40   94 0071E          CLRB    BDATA-7[R0]
03 00000000G  EF                00  E0 00725          BBS     #0,OPTIMIZING,.+3                       ; 2531
                           0000V  31 0072D          BRW     289$
                        00    8F  9F 00730          PUSHAB  #0                                      ; 2533
                  00000000    8F  DF 00733          PUSHAL  #0
                        5E    8F  9F 00739          PUSHAB  #94
                  00000000    8F  DF 0073C          PUSHAL  #0
                        08    8F  9F 00742          PUSHAB  #8
                        01    8F  9F 00745          PUSHAB  #1
            8103  CF                06  FB 00748          CALLS   #6,ALT_SOURCE
                  5C                50  90 0074D          MOVB    R0,PRE_PROCESS
                           0000V  31 00750          BRW     289$
   00000000G  EF          7E    8F  9A 00753 132$:    MOVZBL  #126,MAX_STRING_ANSWER_LENGTH           ; 2542
              50 00000000G   EF  D0 0075B          MOVL    QTAB_OFFSET,R0                          ; 2543
                 FFFFFFF9GEF40   94 00762          CLRB    BDATA-7[R0]
00V00000000G  EF                00  E1 00769          BBC     #0,OPTIMIZING,134$                      ; 2545
                        00    8F  9F 00771          PUSHAB  #0                                      ; 2547
                  00000000    8F  DF 00774          PUSHAL  #0
                        00    8F  9F 0077A          PUSHAB  #0
                  00000000    8F  DF 0077D          PUSHAL  #0
                        0F    8F  9F 00783          PUSHAB  #15
                        00    8F  9F 00786          PUSHAB  #0
            8103  CF                06  FB 00789          CALLS   #6,ALT_SOURCE
                  5C                50  90 0078E          MOVB    R0,PRE_PROCESS
                           0000V  31 00791 134$:    BRW     289$
   00000000G  EF                20  D0 00794 135$:    MOVL    #32,MAX_STRING_ANSWER_LENGTH            ; 2556
              50 00000000G   EF  D0 0079B          MOVL    QTAB_OFFSET,R0                          ; 2557
                 FFFFFFF9GEF40   94 007A2          CLRB    BDATA-7[R0]
03 00000000G  EF                00  E0 007A9          BBS     #0,OPTIMIZING,.+3                       ; 2559
                           0000V  31 007B1          BRW     289$
                        00    8F  9F 007B4          PUSHAB  #0                                      ; 2561
                  00000000    8F  DF 007B7          PUSHAL  #0
                        81    8F  9F 007BD          PUSHAB  #-127
```

```
                                  000000084G   EF    9F 007C0          PUSHAB  IDATA+132
                                         0B    8F    9F 007C6          PUSHAB  #11
                                         01    8F    9F 007C9          PUSHAB  #1
                                8103     CF          06 FB 007CC          CALLS   #6,ALT_SOURCE
                                         5C          50 90 007D1          MOVB    R0,PRE_PROCESS
                                                  0000V 31 007D4          BRW     289$
                             00000000G   EF       7E    8F 9A 007D7 138$:  MOVZBL  #126,MAX_STRING_ANSWER_LENGTH                  ; 2571
                                         50 00000000G   EF 00 007DF          MOVL    QTAB_OFFSET,R0                                    ; 2572
                                         FFFFFFF9GEF40     94 007E6          CLRB    BDATA-7[R0]
                          03 00000000G   EF             00 E0 007ED          BBS     #0,OPTIMIZING,.+3                                 ; 2574
                                                    0000V 31 007F5          BRW     289$
                                                       5C 94 007F8          CLRB    PRE_PROCESS                                       ; 2576
                                                    0000V 31 007FA          BRW     289$
                                         00000088G   EF    D5 007FD 141$:  TSTL    IDATA+136                                         ; 2586
                                                     00V 15 00803          BLEQ    145$
                                         00000084G   EF    D5 00805          TSTL    IDATA+132
                                                     00V 13 0080B          BEQL    144$
                          00V0000000BG   EF             00 E0 0080D          BBS     #0,VDATA+11,145$
                                         5C          01 90 00815 144$:  MOVB    #1,PRE_PROCESS
                                                  0000V 31 00818          BRW     289$
                                                     5C 94 0081B 145$:  CLRB    PRE_PROCESS
                                                  0000V 31 0081D          BRW     289$
                          50 00000000G   EF          19 C5 00820 147$:  MULL3   #25,QTAB_OFFSET,R0                                 ; 2598
                 FFFFFEF2GEF40 00000000G   EF          04 28 00828          MOVC3   #4,BUCKET_DEFAULT,QTAB-270[R0]
                                         00000000    8F    DF 00835          PUSHAL  #0                                                ; 2604
                             00000000G   EF          01 FB 0083B          CALLS   #1,CALC_BUC_OVERHEAD
                             00000000G   EF          50 D0 00842          MOVL    R0,BUCKET_OVERHEAD
                                         00000084G   EF    D5 00849          TSTL    IDATA+132                                         ; 2609
                                                     00V 12 0084F          BNEQ    149$
                             00000000G   EF 000000E4G   EF    D0 00851          MOVL    IDATA+228,ENTRY_SIZE                              ; 2611
                                                     00V 11 0085C          BRB     156$
                          00V00000013G   EF             00 E1 0085E 149$:  BBC     #0,BDATA+19,154$                                  ; 2617
                                            00000000G   EF    D4 00866          CLRL    ENTRY_SIZE                                        ; 2621
                                                        50 D4 0086C          CLRL    R0                                                ; 2623
                             00000000G   EF          50 D0 0086E 151$:  MOVL    R0,TEMP_INT2
                                         51 00000000G   EF    D0 00875          MOVL    TEMP_INT2,R1                                      ; 2627
                          00V00000000GEF41             00 E1 0087C          BBC     #0,SEGMENT_WANTED[R1],153$
                                         51 00000000G   EF    D0 00885          MOVL    TEMP_INT2,R1                                      ; 2629
                             00000000G   EF 00000000GEF41    C0 0088C          ADDL2   SEGMENT_LENGTH[R1],ENTRY_SIZE
                          D2             50             07 F3 00898 153$:  AOBLEQ  #7,R0,151$
                                                        00V 11 0089C          BRB     156$
                             00000000G   EF 000000D8G   EF    D0 0089E 154$:  MOVL    IDATA+216,ENTRY_SIZE                              ; 2638
                                            00000000    8F    DF 008A9 156$:  PUSHAL  #0                                                ; 2642
                             00000000G   EF          01 FB 008AF          CALLS   #1,CALC_REC_OVERHEAD
                          51 00000000G   EF 00000000G   EF    C1 008B6          ADDL3   BUCKET_OVERHEAD,ENTRY_SIZE,R1             ; 2644
                                         51          50 C0 008C2          ADDL2   RECORD_OVERHEAD,R1
                   00000000G   EF        51 00000200    8F    C7 008C5          DIVL3   #512,R1,MIN_BUCKET
                                         51 00000000G   EF 00000000G   EF    C1 008D1          ADDL3   BUCKET_OVERHEAD,ENTRY_SIZE,R1    ; 2647
                                         51          50 C0 008DD          ADDL2   RECORD_OVERHEAD,R1
                 51              51      00          00 7A 008E0          EMUL    #0,#0,R1,R1
                 51              51      51 00000200    8F    7B 008E5          EDIV    #512,R1,R1,R1
                                                        51 D5 008EE          TSTL    R1
                                                        00V 18 008F0          BGEQ    157$
                                         51 00000200    8F    C0 008F2          ADDL2   #512,R1
                                                        51 D5 008F9 157$:  TSTL    R1
                                                        00V 12 008FB          BNEQ    159$
                                            00000000G   EF    D5 008FD          TSTL    MIN_BUCKET
```

```
                                                00V  12 00903          BNEQ    160$
                                 00000000G      EF   D6 00905 159$:    INCL    MIN_BUCKET                                  ; 2653
                                 51 00000000G   EF   19 C5 00908 160$:  MULL3   #25,QTAB_OFFSET,R1                          ; 2655
                          FFFFFEF6GEF41 00000000G   EF   04 28 00913          MOVC3   #4,MIN_BUCKET,QTAB-266[R1]
                                 50 00000000G   EF   19 C5 00920          MULL3   #25,QTAB_OFFSET,R0                          ; 2657
                                                51 FFFFFEF2GEF40   9E 00928          MOVAB   QTAB-270[R0],R1
                          50 FFFFFEF6GEF40      20   00 EE 00930          EXTV    #0,#32,QTAB-266[R0],R0
                                                50   61 D1 0093A          CMPL    (R1),R0
                                                     03 19 0093D          BLSS    .+3
                                                00000V  31 0093F          BRW     289$
                                                61   50 D0 00942          MOVL    R0,(R1)                                     ; 2659
                                                00000V  31 00945          BRW     289$
                                      07        21 000000DCG  EF   CF 00948 163$:  CASEL   IDATA+220,#33,#7                    ; 2670
                                                00000V      00950          .DISPL  168$
                                                00000V      00952          .DISPL  164$
                                                00000V      00954          .DISPL  164$
                                                00000V      00956          .DISPL  165$
                                                00000V      00958          .DISPL  165$
                                                00000V      0095A          .DISPL  166$
                                                00000V      0095C          .DISPL  166$
                                                00000V      0095E          .DISPL  167$
                                                00V  11 00960          BRB     169$
                                   000000000G   EF   02 D0 00962 164$:    MOVL    #2,MAX_KEY_SIZE                             ; 2676
                                   000000000G   EF   02 D0 00969          MOVL    #2,MIN_KEY_SIZE                             ; 2677
                                                00V  11 00970          BRB     170$
                                   000000000G   EF   04 D0 00972 165$:    MOVL    #4,MAX_KEY_SIZE                             ; 2685
                                   000000000G   EF   04 D0 00979          MOVL    #4,MIN_KEY_SIZE                             ; 2686
                                                00V  11 00980          BRB     170$
                                   000000000G   EF   08 D0 00982 166$:    MOVL    #8,MAX_KEY_SIZE                             ; 2694
                                   000000000G   EF   08 D0 00989          MOVL    #8,MIN_KEY_SIZE                             ; 2695
                                                00V  11 00990          BRB     170$
                                   000000000G   EF   10 D0 00992 167$:    MOVL    #16,MAX_KEY_SIZE                            ; 2703
                                   000000000G   EF   01 D0 00999          MOVL    #1,MIN_KEY_SIZE                             ; 2704
                                                00V  11 009A0          BRB     170$
                                   000000000G   EF   FF 8F 9A 009A2 168$:  MOVZBL  #255,MAX_KEY_SIZE                           ; 2712
                                   000000000G   EF   01 D0 009AA          MOVL    #1,MIN_KEY_SIZE                             ; 2713
                                                00V  11 009B1          BRB     170$
                                                     009B3 169$:
                                      000000E4G EF   D5 009B3 170$:    TSTL    IDATA+228                                   ; 2726
                                                00V  13 009B9          BEQL    173$
                                   000000000G   EF 000000E4G EF   D1 009BB          CMPL    IDATA+228,MAX_KEY_SIZE
                                                00V  18 009C6          BGEQ    173$
                                   000000000G   EF 000000E4G EF   D0 009C8          MOVL    IDATA+228,MAX_KEY_SIZE              ; 2732
                                 51 00000000G   EF   19 C5 009D3 173$:  MULL3   #25,QTAB_OFFSET,R1                          ; 2734
                          FFFFFEF6GEF41 00000000G   EF   04 28 009DB          MOVC3   #4,MIN_KEY_SIZE,QTAB-266[R1]
                                 50 00000000G   EF   19 C5 009E8          MULL3   #25,QTAB_OFFSET,R0                          ; 2735
                          FFFFFEFAGEF40 00000000G   EF   04 28 009F0          MOVC3   #4,MAX_KEY_SIZE,QTAB-262[R0]
                                                   00   8F 9F 009FD          PUSHAB  #0                                      ; 2737
                                      000000000G   EF   9F 00A00          PUSHAB  SEGMENT_NUMBER
                                                   85   8F 9F 00A06          PUSHAB  #-123
                                      000000084G   EF   9F 00A09          PUSHAB  IDATA+132
                                                   0B   8F 9F 00A0F          PUSHAB  #11
                                                   01   8F 9F 00A12          PUSHAB  #1
                                        8103       CF   06 FB 00A15          CALLS   #6,ALT_SOURCE
                                                56      50 90 00A1A          MOVB    R0,RESULT
                                                5C      56 90 00A1D          MOVB    RESULT,PRE_PROCESS                      ; 2740
                                                        50 94 00A20          CLRB    R0                                      ; 2742
```

```
                    51 00000000G  EF                19   C5 00A22           MULL3   #25,QTAB_OFFSET,R1
                                  52 FFFFFEF6GEF41    9E 00A2A           MOVAB   QTAB-266[R1],R2
          62 FFFFFEFAGEF41        20                  00   EC 00A32           CMPV    #0,#32,QTAB-262[R1],(R2)
                                                      00V  13 00A3C           BEQL    175$
                                                           50 96 00A3E           INCB    R0
                                  56                       50 8A 00A40 175$:     BICB2   R0,RESULT
                                  00V                      56 E9 00A43           BLBC    RESULT,177$
                                                           5C 94 00A46           CLRB    PRE_PROCESS                                 ; 2750
                       00000000G  EF                       62 D0 00A48           MOVL    (R2),INPUT_VALUE                            ; 2751
                                  52 00000000G  EF         D0 00A4F           MOVL    QTAB_OFFSET,R2                              ; 2752
                    00000000GEF42 00000000G  EF            D0 00A56           MOVL    INPUT_VALUE,IDATA[R2]
                                  52 00000000G  EF         D0 00A62           MOVL    SEGMENT_NUMBER,R2                           ; 2753
                    00000000GEF42 00000000G  EF            D0 00A69           MOVL    INPUT_VALUE,SEGMENT_LENGTH[R2]
                                                0000V      31 00A75 177$:     BRW     289$
                                     00000084G  EF         D5 00A78 178$:     TSTL    IDATA+132                                   ; 2763
                                                00V        13 00A7E           BEQL    180$
                 00V00000033G  EF                00         E0 00A80           BBS     #0,VDATA+51,181$
                                           00    8F         9F 00A88 180$:     PUSHAB  #0                                          ; 2769
                                     00000000    8F         DF 00A8B           PUSHAL  #0
                                           84    8F         9F 00A91           PUSHAB  #-124
                                     00000084G   EF         9F 00A94           PUSHAB  IDATA+132
                                           0B    8F         9F 00A9A           PUSHAB  #11
                                           01    8F         9F 00A9D           PUSHAB  #1
                              8103   CF                  06   FB 00AA0           CALLS   #6,ALT_SOURCE
                                  5C                      50 90 00AA5           MOVB    R0,PRE_PROCESS
                                                0000V      31 00AA8           BRW     289$
                                                   5C      94 00AAB 181$:     CLRB    PRE_PROCESS                                 ; 2774
                                                0000V      31 00AAD           BRW     289$
                                           01    8F         9F 00AB0 183$:     PUSHAB  #1                                          ; 2785
                       00000000G  EF                       01   FB 00AB3           CALLS   #1,SCAN_DEFINITION
                             50 00000000G  EF              19   C5 00ABA           MULL3   #25,QTAB_OFFSET,R0                          ; 2787
                    FFFFFEF6GEF40 00000000G  EF            04   28 00AC2           MOVC3   #4,LOW_KEY,QTAB-266[R0]
                             50 00000000G  EF              19   C5 00ACF           MULL3   #25,QTAB_OFFSET,R0                          ; 2788
                    FFFFFEFAGEF40 00000000G  EF            04   28 00AD7           MOVC3   #4,HIGH_KEY,QTAB-262[R0]
                             50 00000000G  EF              19   C5 00AE4           MULL3   #25,QTAB_OFFSET,R0                          ; 2790
                                  52 FFFFFEF6GEF40         9E 00AEC           MOVAB   QTAB-266[R0],R2
          62 FFFFFEFAGEF40        20                  00   EC 00AF4           CMPV    #0,#32,QTAB-262[R0],(R2)
                                                      03   13 00AFE           BEQL    .+3
                                                0000V      31 00B00           BRW     289$
                                                   5C      94 00B03           CLRB    PRE_PROCESS                                 ; 2796
                       00000000G  EF                       62 D0 00B05           MOVL    (R2),INPUT_VALUE                            ; 2797
                                                0000V      31 00B0C           BRW     289$
                                     00000084G  EF         D5 00B0F 186$:     TSTL    IDATA+132                                   ; 2811
                                                00V        13 00B15           BEQL    188$
                 00V0000001CG  EF                00         E0 00B17           BBS     #0,VDATA+28,189$
                                           00    8F         9F 00B1F 188$:     PUSHAB  #0                                          ; 2817
                                     00000000    8F         DF 00B22           PUSHAL  #0
                                           89    8F         9F 00B28           PUSHAB  #-119
                                     00000000    8F         DF 00B2B           PUSHAL  #0
                                           0C    8F         9F 00B31           PUSHAB  #12
                                           01    8F         9F 00B34           PUSHAB  #1
                              8103   CF                  06   FB 00B37           CALLS   #6,ALT_SOURCE
                                  5C                      50 90 00B3C           MOVB    R0,PRE_PROCESS
                                                0000V      31 00B3F           BRW     289$
                                                   5C      94 00B42 189$:     CLRB    PRE_PROCESS                                 ; 2822
                                                0000V      31 00B44           BRW     289$
                       00000100   8F 000000E8G  EF         D1 00B47 191$:     CMPL    IDATA+232,#256                              ; 2833
```

```
                                    00V  18 00B52          BGEQ    193$
                 00000000G  EF 000000E8G  EF  D0 00B54          MOVL    IDATA+232,CUR_MAX_FIXED                 ; 2835
                                    00V  11 00B5F          BRB     194$
                 00000000G  EF       FF  8F  9A 00B61 193$:  MOVZBL  #255,CUR_MAX_FIXED                      ; 2839
                      50 00000000G  EF       19  C5 00B69 194$:  MULL3   #25,QTAB_OFFSET,R0                   ; 2841
          FFFFFEFAGEF40 00000000G  EF       04  28 00B71          MOVC3   #4,CUR_MAX_FIXED,QTAB-262[R0]
                                      00  8F  9F 00B7E          PUSHAB  #0                                      ; 2843
                                 00000000  8F  DF 00B81          PUSHAL  #0
                                      8A  8F  9F 00B87          PUSHAB  #-118
                                 00000000  8F  DF 00B8A          PUSHAL  #0
                                      0C  8F  9F 00B90          PUSHAB  #12
                                      01  8F  9F 00B93          PUSHAB  #1
                            8103  CF       06  FB 00B96          CALLS   #6,ALT_SOURCE
                                  5C       50  90 00B9B          MOVB    R0,PRE_PROCESS
                                        0000V  31 00B9E          BRW     289$
                                      00  8F  9F 00BA1 195$:  PUSHAB  #0                                      ; 2851
                                 00000007  8F  DF 00BA4          PUSHAL  #7
                                      87  8F  9F 00BAA          PUSHAB  #-121
                                 00000084G  EF  9F 00BAD          PUSHAB  IDATA+132
                                      0B  8F  9F 00BB3          PUSHAB  #11
                                      01  8F  9F 00BB6          PUSHAB  #1
                            8103  CF       06  FB 00BB9          CALLS   #6,ALT_SOURCE
                                  5C       50  90 00BBE          MOVB    R0,PRE_PROCESS
                                        0000V  31 00BC1          BRW     289$
                                 000000C0G  EF  D5 00BC4 196$:  TSTL    IDATA+192                               ; 2863
                                    00V  15 00BCA          BLEQ    198$
                                      00  8F  9F 00BCC          PUSHAB  #0                                      ; 2867
                                 00000000  8F  DF 00BCF          PUSHAL  #0
                                      79  8F  9F 00BD5          PUSHAB  #121
                                 00000084G  EF  9F 00BD8          PUSHAB  IDATA+132
                                      0B  8F  9F 00BDE          PUSHAB  #11
                                      01  8F  9F 00BE1          PUSHAB  #1
                            8103  CF       06  FB 00BE4          CALLS   #6,ALT_SOURCE
                                  5C       50  90 00BE9          MOVB    R0,PRE_PROCESS
                                        0000V  31 00BEC          BRW     289$
                 000000ACG  EF       64  8F  9A 00BEF 198$:  MOVZBL  #100,IDATA+172                          ; 2877
                 00000000G  EF       64  8F  9A 00BF7          MOVZBL  #100,IDATA                              ; 2878
                                       5C  94 00BFF          CLRB    PRE_PROCESS                             ; 2879
                                        0000V  31 00C01          BRW     289$
                 000000E8G  EF 00000000G  EF  D1 00C04 200$:  CMPL    CUR_MAX_REC,IDATA+232                   ; 2894
                                    00V  18 00C0F          BGEQ    202$
                 00000000G  EF 00000000G  EF  D0 00C11          MOVL    CUR_MAX_REC,LOWMAX                      ; 2896
                                    00V  11 00C1C          BRB     203$
                 00000000G  EF 000000E8G  EF  D0 00C1E 202$:  MOVL    IDATA+232,LOWMAX                        ; 2900
                      50 00000000G  EF       19  C5 00C29 203$:  MULL3   #25,QTAB_OFFSET,R0                   ; 2902
          FFFFFEF6GEF40 00000000G  EF       04  28 00C31          MOVC3   #4,LOWMAX,QTAB-266[R0]
                      50 00000000G  EF       19  C5 00C3E          MULL3   #25,QTAB_OFFSET,R0                  ; 2903
          FFFFFEFAGEF40 00000000G  EF       04  28 00C46          MOVC3   #4,CUR_MAX_REC,QTAB-262[R0]
                                      00  8F  9F 00C53          PUSHAB  #0                                      ; 2905
                                 00000000  8F  DF 00C56          PUSHAL  #0
                                      8C  8F  9F 00C5C          PUSHAB  #-116
                                 00000000  8F  DF 00C5F          PUSHAL  #0
                                      0C  8F  9F 00C65          PUSHAB  #12
                                      01  8F  9F 00C68          PUSHAB  #1
                            8103  CF       06  FB 00C6B          CALLS   #6,ALT_SOURCE
                                  5C       50  90 00C70          MOVB    R0,PRE_PROCESS
                                        0000V  31 00C73          BRW     289$
```

```
                                00000000   8F    DF 00C76 204$:   PUSHAL  #0                                        ; 2911
                    00000000G   EF            01    FB 00C7C         CALLS   #1,CALC_BUC_OVERHEAD
                                52            50    D0 00C83         MOVL    R0,R2
                                00000000   8F    DF 00C86         PUSHAL  #0
                    00000000G   EF            01    FB 00C8C         CALLS   #1,CALC_REC_OVERHEAD
                                50          6042    9E 00C93         MOVAB   (R0)[R2],R0
          00000000G EF 00007E00 8F            50    C3 00C97         SUBL3   R0,#32256,CUR_MAX_REC
                                          0000V   31 00CA3         BRW     289$
                    06             00 00000108G  EF    CF 00CA6 205$:   CASEL   IDATA+264,#0,#6                     ; 2921
                                          0000V      00CAE         .DISPL  206$
                                          0000V      00CB0         .DISPL  206$
                                          0000V      00CB2         .DISPL  206$
                                          0000V      00CB4         .DISPL  208$
                                          0000V      00CB6         .DISPL  207$
                                          0000V      00CB8         .DISPL  206$
                                          0000V      00CBA         .DISPL  206$
                                            00V   11 00CBC         BRB     212$
                                00000000   8F    DF 00CBE 206$:   PUSHAL  #0                                        ; 2920
                    00000000G   EF            01    FB 00CC4         CALLS   #1,CALC_BUC_OVERHEAD
                                52            50    D0 00CCB         MOVL    R0,R2
                                00000000   8F    DF 00CCE         PUSHAL  #0
                    00000000G   EF            01    FB 00CD4         CALLS   #1,CALC_REC_OVERHEAD
                                50          6042    9E 00CDB         MOVAB   (R0)[R2],R0
          00000000G EF 00007E00 8F            50    C3 00CDF         SUBL3   R0,#32256,CUR_MAX_REC
                                            00V   11 00CEB         BRB     213$
                    00000000G   EF      7FFF  8F    3C 00CED 207$:   MOVZWL  #32767,CUR_MAX_REC                    ; 2930
                                            00V   11 00CF6         BRB     213$
          00V00000000G          EF            00    E1 00CF8 208$:   BBC     #0,VARIABLE_RECORDS,210$             ; 2931
                    00000000G   EF      3FFD  8F    3C 00D00         MOVZWL  #16381,CUR_MAX_REC                   ; 2933
                                            00V   11 00D09         BRB     211$
                    00000000G   EF      3FFF  8F    3C 00D0B 210$:   MOVZWL  #16383,CUR_MAX_REC                   ; 2937
                                            00V   11 00D14 211$:   BRB     213$
                                                      00D16 212$:
                              04 00000108G  EF    D1 00D16 213$:   CMPL    IDATA+264,#4                         ; 2945
                                            00V   12 00D1D         BNEQ    219$
          00V00000011G          EF            00    E0 00D1F         BBS     #0,BDATA+17,219$
          00V00000000G          EF            00    E1 00D27         BBC     #0,VARIABLE_RECORDS,217$             ; 2951
                    00000000G   EF      01FE  8F    3C 00D2F         MOVZWL  #510,CUR_MAX_REC                     ; 2953
                                            00V   11 00D38         BRB     218$
                    00000000G   EF      0200  8F    3C 00D3A 217$:   MOVZWL  #512,CUR_MAX_REC                     ; 2957
                                                      00D43 218$:
          50 00000000G          EF            19    C5 00D43 219$:   MULL3   #25,QTAB_OFFSET,R0                   ; 2959
FFFFFEFAGEF40 00000000G         EF            04    28 00D4B         MOVC3   #4,CUR_MAX_REC,QTAB-262[R0]
          00V00000000G          EF            00    E1 00D58         BBC     #0,VARIABLE_RECORDS,221$             ; 2961
                                      01      8F    9F 00D60         PUSHAB  #1                                   ; 2963
                                00000000      8F    DF 00D63         PUSHAL  #0
                                      16      8F    9F 00D69         PUSHAB  #22
                                00000000      8F    DF 00D6C         PUSHAL  #0
                                      04      8F    9F 00D72         PUSHAB  #4
                                      01      8F    9F 00D75         PUSHAB  #1
                          8103  CF            06    FB 00D78         CALLS   #6,ALT_SOURCE
                                5C            50    90 00D7D         MOVB    R0,PRE_PROCESS
                                            00V   11 00D80         BRB     222$
                                      00      8F    9F 00D82 221$:   PUSHAB  #0                                   ; 2968
                                00000000      8F    DF 00D85         PUSHAL  #0
                                      8C      8F    9F 00D8B         PUSHAB  #-116
                                00000000      8F    DF 00D8E         PUSHAL  #0
```

```
                                 0C   8F   9F 00D94          PUSHAB  #12
                                 01   8F   9F 00D97          PUSHAB  #1
                          8103   CF         06 FB 00D9A      CALLS   #6,ALT_SOURCE
                                 5C         50 90 00D9F      MOVB    R0,PRE_PROCESS
                                       0000V 31 00DA2 222$:  BRW     289$
                00V00000000G     EF         00 E1 00DA5 223$:  BBC     #0,ISAM_ORG,226$                        ; 2980
                                    00000084G EF    D5 00DAD      TSTL    IDATA+132
                                       00V    13 00DB3      BEQL    226$
                00V00000035G     EF         00 E0 00DB5      BBS     #0,VDATA+53,227$
                                       00   8F   9F 00DBD 226$:  PUSHAB  #0                                    ; 2988
                                 00000000   8F   DF 00DC0      PUSHAL  #0
                                       8B   8F   9F 00DC6      PUSHAB  #-117
                                 00000000   8F   DF 00DC9      PUSHAL  #0
                                       0C   8F   9F 00DCF      PUSHAB  #12
                                       01   8F   9F 00DD2      PUSHAB  #1
                          8103   CF         06 FB 00DD5      CALLS   #6,ALT_SOURCE
                                 5C         50 90 00DDA      MOVB    R0,PRE_PROCESS
                                       0000V 31 00DDD      BRW     289$
                                       5C    94 00DE0 227$:  CLRB    PRE_PROCESS                             ; 2992
                                       0000V 31 00DE2      BRW     289$
             03 00000000G        EF         00 E1 00DE5 229$:  BBC     #0,AUTO_TUNE,.+3                        ; 3000
                                       0000V 31 00DED      BRW     287$
                                 00000001   8F   DF 00DF0      PUSHAL  #1                                      ; 3007
                00000000G        EF         01 FB 00DF6      CALLS   #1,CLEAR
             00V00000000G        EF         00 E1 00DFD      BBC     #0,REGIS,234$                           ; 3012
                                    FFFF9821 EF   9F 00E05      PUSHAB  C.ALZ                                   ; 3016
                                       04   DD 00E0B      PUSHL   #4
                                 00000000G  EF   9F 00E0D      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00E13      CALLS   #3,PAS$WRITE_STRING
                                 05 00000118G EF   D1 00E1A      CMPL    IDATA+280,#5                            ; 3018
                                       00V    13 00E21      BEQL    233$
                                    FFFF9807 EF   9F 00E23      PUSHAB  C.AMA                                   ; 3020
                                    00000046 8F   DD 00E29      PUSHL   #70
                                 00000000G  EF   9F 00E2F      PUSHAB  PAS$FV_OUTPUT
             00000000G           EF         03 FB 00E35      CALLS   #3,PAS$WRITE_STRING
                                    FFFF9836 EF   9F 00E3C 233$:  PUSHAB  C.AMB                                   ; 3023
                                       13   DD 00E42      PUSHL   #19
                                 00000000G  EF   9F 00E44      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00E4A      CALLS   #3,PAS$WRITE_STRING
                                 00000000G  EF   9F 00E51      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         01 FB 00E57      CALLS   #1,PAS$WRITELN2
                                 00000000G  EF   9F 00E5E 234$:  PUSHAB  LOW_SHIFT                               ; 3027
                                       03   DD 00E64      PUSHL   #3
                                 00000000G  EF   9F 00E66      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00E6C      CALLS   #3,PAS$WRITE_STRING
                                       01   DD 00E73      PUSHL   #1
                                       20   DD 00E75      PUSHL   #32
                                 00000000G  EF   9F 00E77      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00E7D      CALLS   #3,PAS$WRITE_CHAR
                                    FFFF9802 EF   9F 00E84      PUSHAB  C.AMC                                   ; 3029
                                       18   DD 00E8A      PUSHL   #24
                                 00000000G  EF   9F 00E8C      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00E92      CALLS   #3,PAS$WRITE_STRING
                                       01   DD 00E99      PUSHL   #1
                                    000000F8G EF   DD 00E9B      PUSHL   IDATA+248
                                 00000000G  EF   9F 00EA1      PUSHAB  PAS$FV_OUTPUT
                00000000G        EF         03 FB 00EA7      CALLS   #3,PAS$WRITE_INTEGER
```

```
                                 01    DD 00EAE          PUSHL   #1
                                 20    DD 00EB0          PUSHL   #32
                      00000000G  EF    9F 00EB2          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00EB8    CALLS   #3,PAS$WRITE_CHAR
                      FFFF97DF   EF    9F 00EBF          PUSHAB  C.AMD                                   ; 3030
                                 06    DD 00EC5          PUSHL   #6
                      00000000G  EF    9F 00EC7          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00ECD    CALLS   #3,PAS$WRITE_STRING
                                 03    DD 00ED4          PUSHL   #3
                      00000084G  EF    DD 00ED6          PUSHL   IDATA+132
                      00000000G  EF    9F 00EDC          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00EE2    CALLS   #3,PAS$WRITE_INTEGER
                      FFFF97BD   EF    9F 00EE9          PUSHAB  C.AME
                                 09    DD 00EEF          PUSHL   #9
                      00000000G  EF    9F 00EF1          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00EF7    CALLS   #3,PAS$WRITE_STRING
          07          21 000000DCG   EF    CF 00EFE          CASEL   IDATA+220,#33,#7                  ; 3034
                           0000V           00F06          .DISPL  242$
                           0000V           00F08          .DISPL  239$
                           0000V           00F0A          .DISPL  235$
                           0000V           00F0C          .DISPL  240$
                           0000V           00F0E          .DISPL  236$
                           0000V           00F10          .DISPL  241$
                           0000V           00F12          .DISPL  237$
                           0000V           00F14          .DISPL  238$
                           0000V     31    00F16          BRW     243$
                      FFFF9799   EF    9F 00F19 235$:     PUSHAB  C.AMF                                   ; 3036
                                 08    DD 00F1F          PUSHL   #8
                      00000000G  EF    9F 00F21          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F27    CALLS   #3,PAS$WRITE_STRING
                           0000V     31    00F2E          BRW     244$
                      FFFF9789   EF    9F 00F31 236$:     PUSHAB  C.AMG                                   ; 3037
                                 08    DD 00F37          PUSHL   #8
                      00000000G  EF    9F 00F39          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F3F    CALLS   #3,PAS$WRITE_STRING
                           0000V     31    00F46          BRW     244$
                      FFFF9779   EF    9F 00F49 237$:     PUSHAB  C.AMH                                   ; 3038
                                 08    DD 00F4F          PUSHL   #8
                      00000000G  EF    9F 00F51          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F57    CALLS   #3,PAS$WRITE_STRING
                                 00V   11    00F5E          BRB     244$
                      FFFF976A   EF    9F 00F60 238$:     PUSHAB  C.AMI                                   ; 3039
                                 08    DD 00F66          PUSHL   #8
                      00000000G  EF    9F 00F68          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F6E    CALLS   #3,PAS$WRITE_STRING
                                 00V   11    00F75          BRB     244$
                      FFFF975B   EF    9F 00F77 239$:     PUSHAB  C.AMJ                                   ; 3040
                                 08    DD 00F7D          PUSHL   #8
                      00000000G  EF    9F 00F7F          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F85    CALLS   #3,PAS$WRITE_STRING
                                 00V   11    00F8C          BRB     244$
                      FFFF974C   EF    9F 00F8E 240$:     PUSHAB  C.AMK                                   ; 3041
                                 08    DD 00F94          PUSHL   #8
                      00000000G  EF    9F 00F96          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF          03    FB 00F9C    CALLS   #3,PAS$WRITE_STRING
                                 00V   11    00FA3          BRB     244$
                      FFFF973D   EF    9F 00FA5 241$:     PUSHAB  C.AML                                   ; 3042
```

```
                    08   DD 00FAB          PUSHL   #8
                00000000G EF   9F 00FAD      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 00FB3      CALLS   #3,PAS$WRITE_STRING
                           00V 11   00FBA      BRB     244$
                FFFF972E   EF   9F 00FBC 242$: PUSHAB  C.AMM                    ; 3043
                    08   DD 00FC2          PUSHL   #8
                00000000G EF   9F 00FC4      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 00FCA      CALLS   #3,PAS$WRITE_STRING
                           00V 11   00FD1      BRB     244$
                                00FD3 243$:
                FFFF971F   EF   9F 00FD3 244$: PUSHAB  C.AMN                    ; 3051
                    0D   DD 00FD9          PUSHL   #13
                00000000G EF   9F 00FDB      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 00FE1      CALLS   #3,PAS$WRITE_STRING
                05 00000118G EF   D1 00FE8      CMPL    IDATA+280,#5            ; 3053
                           00V 13   00FEF      BEQL    246$
                FFFF9711   EF   9F 00FF1      PUSHAB  C.AMO                   ; 3055
                    05   DD 00FF7          PUSHL   #5
                00000000G EF   9F 00FF9      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 00FFF      CALLS   #3,PAS$WRITE_STRING
                00000098G EF   D5 01006 246$: TSTL    IDATA+152               ; 3057
                           00V 12   0100C      BNEQ    248$
                FFFF96FC   EF   9F 0100E      PUSHAB  C.AMP                   ; 3059
                    07   DD 01014          PUSHL   #7
                00000000G EF   9F 01016      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 0101C      CALLS   #3,PAS$WRITE_STRING
                           00V 11   01023      BRB     249$
                FFFF96ED   EF   9F 01025 248$: PUSHAB  C.AMQ                   ; 3063
                    07   DD 0102B          PUSHL   #7
                00000000G EF   9F 0102D      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 01033      CALLS   #3,PAS$WRITE_STRING
                05 00000118G EF   D1 0103A 249$: CMPL    IDATA+280,#5            ; 3065
                           00V 12   01041      BNEQ    251$
00000000G EF                   00   FB 01043      CALLS   #0,NATURAL_DEPTH        ; 3069
00000000G EF                   50   D0 0104A      MOVL    R0,BUCKET_DEFAULT
                    FFFF96C9   EF   9F 01051      PUSHAB  C.AMR                   ; 3070
                    02   DD 01057          PUSHL   #2
                00000000G EF   9F 01059      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 0105F      CALLS   #3,PAS$WRITE_STRING
                    02   DD 01066          PUSHL   #2
                00000000G EF   DD 01068      PUSHL   BUCKET_DEFAULT
                00000000G EF   9F 0106E      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 01074      CALLS   #3,PAS$WRITE_INTEGER
                    01   DD 0107B          PUSHL   #1
                    29   DD 0107D          PUSHL   #41
                00000000G EF   9F 0107F      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 01085      CALLS   #3,PAS$WRITE_CHAR
                00000000G EF   9F 0108C 251$: PUSHAB  PAS$FV_OUTPUT           ; 3074
00000000G EF                   01   FB 01092      CALLS   #1,PAS$WRITELN2
                00000000G EF   9F 01099      PUSHAB  LOW_SHIFT               ; 3075
                    03   DD 0109F          PUSHL   #3
                00000000G EF   9F 010A1      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 010A7      CALLS   #3,PAS$WRITE_STRING
                    01   DD 010AE          PUSHL   #1
                    20   DD 010B0          PUSHL   #32
                00000000G EF   9F 010B2      PUSHAB  PAS$FV_OUTPUT
00000000G EF                   03   FB 010B8      CALLS   #3,PAS$WRITE_CHAR
```

```
                    FFFF965F    EF   9F 010BF          PUSHAB  C.AMS                                   ; 3077
                                0A   DD 010C5          PUSHL   #10
                    00000000G   EF   9F 010C7          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 010CD       CALLS   #3,PAS$WRITE_STRING
                                    03   DD 010D4       PUSHL   #3
                    00000084G   EF   DD 010D6          PUSHL   IDATA+132
                    00000000G   EF   9F 010DC          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 010E2       CALLS   #3,PAS$WRITE_INTEGER
                    FFFF9641    EF   9F 010E9          PUSHAB  C.AMT
                                09   DD 010EF          PUSHL   #9
                    00000000G   EF   9F 010F1          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 010F7       CALLS   #3,PAS$WRITE_STRING
00V00000017G      EF              00   E1 010FE       BBC     #0,BDATA+23,253$                        ; 3081
                    FFFF9630    EF   9F 01106          PUSHAB  C.AMU                                   ; 3083
                                04   DD 0110C          PUSHL   #4
                    00000000G   EF   9F 0110E          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 01114       CALLS   #3,PAS$WRITE_STRING
                                    00V  11 0111B       BRB     254$
                    FFFF961D    EF   9F 0111D 253$:    PUSHAB  C.AMV                                   ; 3087
                                04   DD 01123          PUSHL   #4
                    00000000G   EF   9F 01125          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 0112B       CALLS   #3,PAS$WRITE_STRING
                 05 00000118G   EF   D1 01132 254$:    CMPL    IDATA+280,#5                            ; 3089
                                    00V  13 01139       BEQL    256$
                 04 00000118G   EF   D1 0113B          CMPL    IDATA+280,#4
                                    00V  13 01142       BEQL    257$
                    FFFF95FA    EF   9F 01144 256$:    PUSHAB  C.AMW                                   ; 3092
                                06   DD 0114A          PUSHL   #6
                    00000000G   EF   9F 0114C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 01152       CALLS   #3,PAS$WRITE_STRING
                                    03   DD 01159       PUSHL   #3
                    00000084G   EF   DD 0115B          PUSHL   IDATA+132
                    00000000G   EF   9F 01161          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 01167       CALLS   #3,PAS$WRITE_INTEGER
                    FFFF95D8    EF   9F 0116E          PUSHAB  C.AMX
                                0D   DD 01174          PUSHL   #13
                    00000000G   EF   9F 01176          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 0117C       CALLS   #3,PAS$WRITE_STRING
                                    03   DD 01183       PUSHL   #3
                    000000D8G   EF   DD 01185          PUSHL   IDATA+216
                    00000000G   EF   9F 0118B          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 01191       CALLS   #3,PAS$WRITE_INTEGER
                                    01   DD 01198       PUSHL   #1
                                    20   DD 0119A       PUSHL   #32
                    00000000G   EF   9F 0119C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 011A2       CALLS   #3,PAS$WRITE_CHAR
                    FFFF95AD    EF   9F 011A9 257$:    PUSHAB  C.AMY                                   ; 3095
                                06   DD 011AF          PUSHL   #6
                    00000000G   EF   9F 011B1          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 011B7       CALLS   #3,PAS$WRITE_STRING
                                    03   DD 011BE       PUSHL   #3
                    00000084G   EF   DD 011C0          PUSHL   IDATA+132
                    00000000G   EF   9F 011C6          PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 011CC       CALLS   #3,PAS$WRITE_INTEGER
                    FFFF958B    EF   9F 011D3          PUSHAB  C.AMZ
                                0B   DD 011D9          PUSHL   #11
                    00000000G   EF   9F 011DB          PUSHAB  PAS$FV_OUTPUT
```

```
                 00000000G  EF                03   FB 011E1          CALLS   #3,PAS$WRITE_STRING
                                              05   DD 011E8          PUSHL   #5
                            000000CCG         EF   DD 011EA          PUSHL   IDATA+204
                            00000000G         EF   9F 011F0          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 011F6          CALLS   #3,PAS$WRITE_INTEGER
                                              01   DD 011FD          PUSHL   #1
                                              20   DD 011FF          PUSHL   #32
                            00000000G         EF   9F 01201          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 01207          CALLS   #3,PAS$WRITE_CHAR
                            00000000G         EF   9F 0120E          PUSHAB  PAS$FV_OUTPUT                   ; 3098
                 00000000G  EF                01   FB 01214          CALLS   #1,PAS$WRITELN2
                            00000000G         EF   9F 0121B          PUSHAB  LOW_SHIFT                       ; 3099
                                              03   DD 01221          PUSHL   #3
                            00000000G         EF   9F 01223          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 01229          CALLS   #3,PAS$WRITE_STRING
                                              01   DD 01230          PUSHL   #1
                                              20   DD 01232          PUSHL   #32
                            00000000G         EF   9F 01234          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 0123A          CALLS   #3,PAS$WRITE_CHAR
                            02 000000F8G      EF   D1 01241          CMPL    IDATA+248,#2                    ; 3101
                                              03   14 01248          BGTR    .+3
                                        0000V 31   24 0124A          BRW     259$
                            FFFF951D          EF   9F 0124D          PUSHAB  C.ANA                           ; 3105
                                              15   DD 01253          PUSHL   #21
                            00000000G         EF   9F 01255          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 0125B          CALLS   #3,PAS$WRITE_STRING
                                              03   DD 01262          PUSHL   #3
          50 00000024G  EF  000043C8          8F   45 01264          MULF3   #^F100.0,RDATA+36,R0
                            7E                50   4A 01270          CVTFL   R0,-(SP)
                            00000000G         EF   9F 01273          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 01279          CALLS   #3,PAS$WRITE_INTEGER
                            FFFF9500          EF   9F 01280          PUSHAB  C.ANB
                                              02   DD 01286          PUSHL   #2
                            00000000G         EF   9F 01288          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 0128E          CALLS   #3,PAS$WRITE_STRING
                            FFFF94ED          EF   9F 01295          PUSHAB  C.ANC                           ; 3107
                                              15   DD 0129B          PUSHL   #21
                            00000000G         EF   9F 0129D          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 012A3          CALLS   #3,PAS$WRITE_STRING
                                              03   DD 012AA          PUSHL   #3
          50 00000020G  EF  000043C8          8F   45 012AC          MULF3   #^F100.0,RDATA+32,R0
                            7E                50   4A 012B8          CVTFL   R0,-(SP)
                            00000000G         EF   9F 012BB          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 012C1          CALLS   #3,PAS$WRITE_INTEGER
                            FFFF94D0          EF   9F 012C8          PUSHAB  C.AND
                                              02   DD 012CE          PUSHL   #2
                            00000000G         EF   9F 012D0          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 012D6          CALLS   #3,PAS$WRITE_STRING
                            FFFF94BD          EF   9F 012DD          PUSHAB  C.ANE                           ; 3109
                                              15   DD 012E3          PUSHL   #21
                            00000000G         EF   9F 012E5          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 012EB          CALLS   #3,PAS$WRITE_STRING
                                              03   DD 012F2          PUSHL   #3
          50 00000028G  EF  000043C8          8F   45 012F4          MULF3   #^F100.0,RDATA+40,R0
                            7E                50   4A 01300          CVTFL   R0,-(SP)
                            00000000G         EF   9F 01303          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03   FB 01309          CALLS   #3,PAS$WRITE_INTEGER
```

```
                    FFFF94A0   EF   9F 01310          PUSHAB  C.ANF
                               02   DD 01316          PUSHL   #2
                    00000000G  EF   9F 01318          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 0131E          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 01325          PUSHAB  PAS$FV_OUTPUT                          ; 3112
00000000G   EF                 01   FB 0132B          CALLS   #1,PAS$WRITELN2
                    00000000G  EF   9F 01332          PUSHAB  LOW_SHIFT                              ; 3113
                               03   DD 01338          PUSHL   #3
                    00000000G  EF   9F 0133A          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 01340          CALLS   #3,PAS$WRITE_STRING
                               01   DD 01347          PUSHL   #1
                               20   DD 01349          PUSHL   #32
                    00000000G  EF   9F 0134B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 01351          CALLS   #3,PAS$WRITE_CHAR
              05    00000118G  EF   D1 01358 259$:    CMPL    IDATA+280,#5                           ; 3117
                               00V  13 0135F          BEQL    261$
                    00000118G  EF   D5 01361          TSTL    IDATA+280
                               00V  13 01367          BEQL    262$
                    FFFF9449   EF   9F 01369 261$:    PUSHAB  C.ANG                                  ; 3120
                               15   DD 0136F          PUSHL   #21
                    00000000G  EF   9F 01371          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 01377          CALLS   #3,PAS$WRITE_STRING
                               03   DD 0137E          PUSHL   #3
                    000000ACG  EF   DD 01380          PUSHL   IDATA+172
                    00000000G  EF   9F 01386          PUSHAB  PAS$FV_OUTPUT
000000000G  EF                 03   FB 0138C          CALLS   #3,PAS$WRITE_INTEGER
                    FFFF9435   EF   9F 01393          PUSHAB  C.ANH
                               02   DD 01399          PUSHL   #2
                    00000000G  EF   9F 0139B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 013A1          CALLS   #3,PAS$WRITE_STRING
                    FFFF9422   EF   9F 013A8 262$:    PUSHAB  C.ANI                                  ; 3122
                               11   DD 013AE          PUSHL   #17
                    00000000G  EF   9F 013B0          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 013B6          CALLS   #3,PAS$WRITE_STRING
00V00000000G EF                00   E1 013BD          BBC     #0,VARIABLE_RECORDS,264$               ; 3124
                    FFFF9419   EF   9F 013C5          PUSHAB  C.ANJ                                  ; 3126
                               09   DD 013CB          PUSHL   #9
                    00000000G  EF   9F 013CD          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 013D3          CALLS   #3,PAS$WRITE_STRING
                               00V  11 013DA          BRB     265$
                    FFFF940E   EF   9F 013DC 264$:    PUSHAB  C.ANK                                  ; 3130
                               09   DD 013E2          PUSHL   #9
                    00000000G  EF   9F 013E4          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 013EA          CALLS   #3,PAS$WRITE_STRING
              05    00000118G  EF   D1 01[illegible]1 265$:    CMPL    IDATA+280,#5                   ; 3132
                               00V  13 013F3          BEQL    267$
              01    00000118G  EF   D1 013FA          CMPL    IDATA+280,#1
                               00V  13 01401          BEQL    271$
                    FFFF93F3   EF   9F 01403 267$:    PUSHAB  C.ANL                                  ; 3137
                               03   DD 01409          PUSHL   #3
                    00000000G  EF   9F 0140B          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 01411          CALLS   #3,PAS$WRITE_STRING
00V00000000G EF                00   E1 01418          BBC     #0,VARIABLE_RECORDS,269$               ; 3139
                    FFFF93DA   EF   9F 01420          PUSHAB  C.ANM                                  ; 3141
                               11   DD 01426          PUSHL   #17
                    00000000G  EF   9F 01428          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                 03   FB 0142E          CALLS   #3,PAS$WRITE_STRING
```

```
                                 00V  11 01435          BRB     270$
                        FFFF93D7 EF   9F 01437 269$:    PUSHAB  C.ANN                                   ; 3145
                                 11   DD 0143D          PUSHL   #17
                        00000000G EF  9F 0143F          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 01445          CALLS   #3,PAS$WRITE_STRING
                                 05   DD 0144C 270$:     PUSHL   #5                                      ; 3147
                        000000E8G EF  DD 0144E          PUSHL   IDATA+232
                        00000000G EF  9F 01454          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 0145A          CALLS   #3,PAS$WRITE_INTEGER
                                 01   DD 01461          PUSHL   #1
                                 20   DD 01463          PUSHL   #32
                        00000000G EF  9F 01465          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 0146B          CALLS   #3,PAS$WRITE_CHAR
                        00000000G EF  9F 01472 271$:     PUSHAB  PAS$FV_OUTPUT                           ; 3152
          00000000G EF           01   FB 01478          CALLS   #1,PAS$WRITELN2
                        00000000G EF  9F 0147F          PUSHAB  LOW_SHIFT                               ; 3153
                                 03   DD 01485          PUSHL   #3
                        00000000G EF  9F 01487          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 0148D          CALLS   #3,PAS$WRITE_STRING
                                 01   DD 01494          PUSHL   #1
                                 20   DD 01496          PUSHL   #32
                        00000000G EF  9F 01498          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 0149E          CALLS   #3,PAS$WRITE_CHAR
                        FFFF937D EF   9F 014A5          PUSHAB  C.ANO                                   ; 3155
                                 0F   DD 014AB          PUSHL   #15
                        00000000G EF  9F 014AD          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 014B3          CALLS   #3,PAS$WRITE_STRING
02                  00 000000E0G EF   CF 014BA          CASEL   IDATA+224,#0,#2                         ; 3157
                                0000V     014C2          .DISPL  272$
                                0000V     014C4          .DISPL  273$
                                0000V     014C6          .DISPL  274$
                                  00V  11 014C8          BRB     275$
                        FFFF9368 EF   9F 014CA 272$:    PUSHAB  C.ANP                                   ; 3159
                                 0B   DD 014D0          PUSHL   #11
                        00000000G EF  9F 014D2          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 014D8          CALLS   #3,PAS$WRITE_STRING
                                 00V  11 014DF          BRB     276$
                        FFFF935D EF   9F 014E1 273$:    PUSHAB  C.ANQ                                   ; 3160
                                 0B   DD 014E7          PUSHL   #11
                        00000000G EF  9F 014E9          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 014EF          CALLS   #3,PAS$WRITE_STRING
                                 00V  11 014F6          BRB     276$
                        FFFF9352 EF   9F 014F8 274$:    PUSHAB  C.ANR                                   ; 3161
                                 0B   DD 014FE          PUSHL   #11
                        00000000G EF  9F 01500          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 01506          CALLS   #3,PAS$WRITE_STRING
                                 00V  11 015CD          BRB     276$
                                      0150F 275$:
                     05 00000118G EF  D1 0150F 276$:    CMPL    IDATA+280,#5                            ; 3169
                                 00V  13 01516          BEQL    278$
                     02 00000118G EF  D1 01518          CMPL    IDATA+280,#2
                                 00V  13 0151F          BEQL    279$
                        FFFF9335 EF   9F 01521 278$:    PUSHAB  C.ANS                                   ; 3172
                                 10   DD 01527          PUSHL   #16
                        00000000G EF  9F 01529          PUSHAB  PAS$FV_OUTPUT
          00000000G EF           03   FB 0152F          CALLS   #3,PAS$WRITE_STRING
                                 09   DD 01536          PUSHL   #9
```

```
                    000000C0G  EF    DD 01538          PUSHL   IDATA+192
                    00000000G  EF    9F 0153E          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 01544          CALLS   #3,PAS$WRITE_INTEGER
                               01    DD 0154B          PUSHL   #1
                               20    DD 0154D          PUSHL   #32
                    00000000G  EF    9F 0154F          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 01555          CALLS   #3,PAS$WRITE_CHAR
                         05 00000118G EF  D1 0155C 279$: CMPL    IDATA+280,#5                    ; 3174
                               00V   13 01563          BEQL    281$
                         03 00000118G EF  D1 01565      CMPL    IDATA+280,#3
                               00V   13 0156C          BEQL    282$
                       FFFF92F8    EF    9F 0156E 281$: PUSHAB  C.ANT                           ; 3177
                               10    DD 01574          PUSHL   #16
                    00000000G  EF    9F 01576          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 0157C          CALLS   #3,PAS$WRITE_STRING
                               09    DD 01583          PUSHL   #9
                    00000088G  EF    DD 01585          PUSHL   IDATA+136
                    00000000G  EF    9F 01588          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 01591          CALLS   #3,PAS$WRITE_INTEGER
                               01    DD 01598          PUSHL   #1
                               20    DD 0159A          PUSHL   #32
                    00000000G  EF    9F 0159C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 015A2          CALLS   #3,PAS$WRITE_CHAR
00V000000000G EF                   00 E1 015A9 282$: BBC     #0,REGIS,284$                   ; 3182
                       FFFF92C5    EF    9F 015B1          PUSHAB  C.ANU                          ; 3184
                               04    DD 015B7          PUSHL   #4
                    00000000G  EF    9F 015B9          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 015BF          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 015C6          PUSHAB  CRLF
                               02    DD 015CC          PUSHL   #2
                    00000000G  EF    9F 015CE          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 015D4          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 015DB          PUSHAB  CRLF
                               02    DD 015E1          PUSHL   #2
                    00000000G  EF    9F 015E3          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 015E9          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 015F0          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               01 FB 015F6       CALLS   #1,PAS$WRITELN2
                    00000000G  EF    9F 015FD 284$:    PUSHAB  PAS$FV_OUTPUT                   ; 3186
          00000000G EF               01 FB 01603       CALLS   #1,PAS$WRITELN2
                         03 000000F8G EF  D1 0160A      CMPL    IDATA+248,#3                    ; 3191
                               00V   18 01611          BGEQ    287$
                    00000000G  EF    9F 01613          PUSHAB  PAS$FV_OUTPUT                   ; 3193
          00000000G EF               01 FB 01619       CALLS   #1,PAS$WRITELN2
                               00V   11 01620 287$:    BRB     289$
                                        01622 288$:
                         50            5C  90 01622 289$: MOVB    PRE_PROCESS,R0               ; 3205
                                        04 01625          RET

; Routine Size: 5670 bytes,    Routine Base: $CODE + 083FC

                                        00000 VERIFY_PROCESS:                                   ; 3250
                                   003C 00000          .WORD   ^M<R2,R3,R4,R5>
                         5C            01  90 00002     MOVB    #1,VERIFY_PROCESS              ; 3257
                   3C          0B 00000000G EF  CF 00005 CASEL   QTAB_OFFSET,#11,#60            ; 3259
                                        0000V     0000D  .DISPL  1$
                                        0000V     0000F  .DISPL  1$
```

```
0000V   00011           .DISPL  1$
0000V   00013           .DISPL  1$
0000V   00015           .DISPL  1$
007A    00017           .DISPL  122
007A    00019           .DISPL  122
007A    0001B           .DISPL  122
007A    0001D           .DISPL  122
007A    0001F           .DISPL  122
007A    00021           .DISPL  122
007A    00023           .DISPL  122
007A    00025           .DISPL  122
007A    00027           .DISPL  122
007A    00029           .DISPL  122
007A    0002B           .DISPL  122
007A    0002D           .DISPL  122
007A    0002F           .DISPL  122
007A    00031           .DISPL  122
007A    00033           .DISPL  122
007A    00035           .DISPL  122                                     ; R
007A    00037           .DISPL  122
007A    00039           .DISPL  122
007A    0003B           .DISPL  122
007A    0003D           .DISPL  122
007A    0003F           .DISPL  122
007A    00041           .DISPL  122
007A    00043           .DISPL  122
007A    00045           .DISPL  122
007A    00047           .DISPL  122
007A    00049           .DISPL  122
0000V   0004B           .DISPL  16$
0000V   0004D           .DISPL  11$
0000V   0004F           .DISPL  11$                                     ; R
0000V   00051           .DISPL  11$
007A    00053           .DISPL  122
007A    00055           .DISPL  122
007A    00057           .DISPL  122
007A    00059           .DISPL  122
007A    0005B           .DISPL  122
007A    0005D           .DISPL  122
007A    0005F           .DISPL  122
007A    00061           .DISPL  122
007A    00063           .DISPL  122
007A    00065           .DISPL  122
007A    00067           .DISPL  122
007A    00069           .DISPL  122                                     ; R
007A    0006B           .DISPL  122
007A    0006D           .DISPL  122
007A    0006F           .DISPL  122
007A    00071           .DISPL  122
007A    00073           .DISPL  122
007A    00075           .DISPL  122
0000V   00077           .DISPL  51$
007A    00079           .DISPL  122
007A    0007B           .DISPL  122
007A    0007D           .DISPL  122
007A    0007F           .DISPL  122
007A    00081           .DISPL  122
```

```
                                    0000V     00083          .DISPL  45$
                                    0000V     00085          .DISPL  57$
                                    0000V  31  00087          BRW     93$
                          50 00000000G  EF  DO  0008A 1$:     MOVL    QTAB_OFFSET,R0                                  ; 3269
                          50 FFFFFFB0GEF40  7E  00091          MOVAQ   SDATA-80[R0],R0
                                       60  B5  00099          TSTW    (R0)
                                       00V 12  0009B          BNEQ    3$
                          51 00000000G  EF  DO  0009D          MOVL    QTAB_OFFSET,R1                                  ; 3273
                             FFFFFFF9GEF41  94  000A4          CLRB    BDATA-7[R1]
                                    0000V  31  000AB          BRW     94$
00000000G  EF        60              10     00  ED  000AE 3$:     CMPZV   #0,#16,(R0),MAX_STRING_ANSWER_LENGTH            ; 3281
                                       00V 15  000B7          BLEQ    8$
                     00V00000000G  EF     00  E1  000B9          BBC     #0,OPTIMIZING,6$                                ; 3287
                          60 00000000G  EF  B0  000C1          MOVW    MAX_STRING_ANSWER_LENGTH,(R0)                   ; 3291
                          51 00000000G  EF  DO  000C8          MOVL    QTAB_OFFSET,R1                                  ; 3293
                FFFFFFF9GEF41           01  90  000CF          MOVB    #1,BDATA-7[R1]
                                    0000V  31  000D7          BRW     94$
                                       50  DD  000DA 6$:     PUSHL   R0                                              ; 3301
                   00000000G  EF        01  FB  000DC          CALLS   #1,STR$FREE1_DX
                                       5C  94  000E3          CLRB    VERIFY_PROCESS                                  ; 3302
                                    0000V  31  000E5          BRW     94$
                          50 00000000G  EF  DO  000E8 8$:     MOVL    QTAB_OFFSET,R0                                  ; 3310
                FFFFFFF9GEF40           01  90  000EF          MOVB    #1,BDATA-7[R0]
                                    0000V  31  000F7          BRW     94$
                          50 00000000G  EF  DO  000FA 11$:    MOVL    QTAB_OFFSET,R0                                  ; 3320
                          50 00000000GEF40  DE  00101          MOVAL   IDATA[R0],R0
                          32              60  D1  00109          CMPL    (R0),#50
                                          00V 18  0010C          BGEQ    15$
                          60              32  DO  0010E          MOVL    #50,(R0)                                        ; 3324
                00V00000000G  EF          00  E0  00111          BBS     #0,AUTO_TUNE,14$                                ; 3326
                                00000000G EF  9F  00119          PUSHAB  SHIFT                                           ; 3330
                                          04  DD  0011F          PUSHL   #4
                                00000000G EF  9F  00121          PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF                  03  FB  00127          CALLS   #3,PAS$WRITE_STRING
                                FFFF9126  EF  9F  0012E          PUSHAB  C.ANV
                                          1B  DD  00134          PUSHL   #27
                                00000000G EF  9F  00136          PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF                  03  FB  0013C          CALLS   #3,PAS$WRITE_STRING
                                00000000G EF  9F  00143          PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF                  01  FB  00149          CALLS   #1,PAS$WRITELN2
                                00004140  8F  DF  00150          PUSHAF  #^F3.0                                          ; 3331
                   00000000G  EF                  01  FB  00156          CALLS   #1,LIB$WAIT
                                                     0015D 14$:
                                          0000V  31  0015D 15$:     BRW     94$
                          05 00000118G  EF  D1  00160 16$:     CMPL    IDATA+280,#5                                    ; 3344
                                          00V 13  00167          BEQL    31$
                                00000118G EF  D5  00169          TSTL    IDATA+280                                       ; 3346
                                          00V 12  0016F          BNEQ    19$
                          2B 00000000G  EF  D1  00171          CMPL    INPUT_VALUE,#43
                                          00V 13  00178          BEQL    29$
                          02 00000118G  EF  D1  0017A 19$:     CMPL    IDATA+280,#2
                                          00V 12  00181          BNEQ    21$
                          30 00000000G  EF  D1  00183          CMPL    INPUT_VALUE,#48
                                          00V 13  0018A          BEQL    29$
                          03 00000118G  EF  D1  0018C 21$:     CMPL    IDATA+280,#3
                                          00V 12  00193          BNEQ    23$
                          22 00000000G  EF  D1  00195          CMPL    INPUT_VALUE,#34
```

```
                                00V  13 0019C          BEQL    29$
                     01 00000118G   EF  D1 0019E 23$:  CMPL    IDATA+280,#1
                                00V  12 001A5          BNEQ    25$
                     3A 00000000G   EF  D1 001A7          CMPL    INPUT_VALUE,#58
                                00V  13 001AE          BEQL    29$
                     04 00000118G   EF  D1 001B0 25$:  CMPL    IDATA+280,#4
                                00V  12 001B7          BNEQ    27$
                     36 00000000G   EF  D1 001B9          CMPL    INPUT_VALUE,#54
                                00V  13 001C0          BEQL    29$
           00000100  8F 000000DCG   EF  D1 001C2 27$:  CMPL    IDATA+220,#256
                                00V  1E 001CD          BGEQU   28$
       00VFFFF909B  EF 000000DCG   EF  E0 001CF          BBS     IDATA+220,C.ANW,31$
                     36 00000000G   EF  D1 001DB 28$:  CMPL    INPUT_VALUE,#54
                                00V  12 001E2          BNEQ    31$
                                5C   94 001E4 29$:  CLRB    VERIFY_PROCESS                  ; 3368
                        00000084G   EF  D5 001E6 31$:  TSTL    IDATA+132                       ; 3374
                                00V  13 001EC          BEQL    34$
           00000100  8F 00000000G   EF  D1 001EE          CMPL    INPUT_VALUE,#256
                                00V  1E 001F9          BGEQU   34$
       00VFFFF908F  EF 00000000G   EF  E1 001FB          BBC     INPUT_VALUE,C.ANX,34$
                                5C   94 00207          CLRB    VERIFY_PROCESS          ; 3379
                     03 000000F8G   EF  D1 00209 34$:  CMPL    IDATA+248,#3            ; 3381
                                00V  18 00210          BGEQ    37$
           00000100  8F 00000000G   EF  D1 00212          CMPL    INPUT_VALUE,#256
                                00V  1E 0021D          BGEQU   37$
       00VFFFF908B  EF 00000000G   EF  E1 0021F          BBC     INPUT_VALUE,C.ANY,37$
                                5C   94 0022B          CLRB    VERIFY_PROCESS          ; 3386
                     01 000000C0G   EF  D1 0022D 37$:  CMPL    IDATA+192,#1            ; 3388
                                00V  18 00234          BGEQ    40$
                     38 00000000G   EF  D1 00236          CMPL    INPUT_VALUE,#56
                                00V  12 0023D          BNEQ    40$
                                5C   94 0023F          CLRB    VERIFY_PROCESS          ; 3391
                     01 000000C0G   EF  D1 00241 40$:  CMPL    IDATA+192,#1            ; 3393
                                00V  18 00248          BGEQ    44$
                     2B 00000000G   EF  D1 0024A          CMPL    INPUT_VALUE,#43
                                00V  12 00251          BNEQ    44$
   00V00000000G  EF                 00  E0 00253          BBS     #0,AUTO_TUNE,44$
                        00000000G   EF  9F 0025B          PUSHAB  SHIFT                   . 3403
                                04   DD 00261          PUSHL   #4
                        00000000G   EF  9F 00263          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF             03   FB 00269          CALLS   #3,PAS$WRITE_STRING
                        00000000G   EF  9F 00270          PUSHAB  ANSI_REVERSE
                                04   DD 00276          PUSHL   #4
                        00000000G   EF  9F 00278          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF             03   FB 0027E          CALLS   #3,PAS$WRITE_STRING
                        FFFF904B    EF  9F 00285          PUSHAB  C.ANZ
                                35   DD 0028B          PUSHL   #53
                        00000000G   EF  9F 0028D          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF             03   FB 00293          CALLS   #3,PAS$WRITE_STRING
                        00000000G   EF  9F 0029A          PUSHAB  ANSI_RESET
                                04   DD 002A0          PUSHL   #4
                        00000000G   EF  9F 002A2          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF             03   FB 002A8          CALLS   #3,PAS$WRITE_STRING
                        00000000G   EF  9F 002AF          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF             01   FB 002B5          CALLS   #1,PAS$WRITELN2
                        00004140    8F  DF 002BC          PUSHAF  #^F3.0                  ; 3406
          00000000G  EF             01   FB 002C2          CALLS   #1,LIB$WAIT
```

```
                                    0000V  31 002C9 44$:    BRW     94$
                          00000084G EF     D5 002CC 45$:    TSTL    IDATA+132                                   ; 3419
                                    00V    13 002D2         BEQL    50$
                       01 00000000G EF     D1 002D4         CMPL    INPUT_VALUE,#1
                                    00V    13 002DB         BEQL    49$
                       02 00000000G EF     D1 002DD         CMPL    INPUT_VALUE,#2
                                    00V    13 002E4         BEQL    49$
                       03 00000000G EF     D1 002E6         CMPL    INPUT_VALUE,#3
                                    00V    12 002ED         BNEQ    50$
                                    5C     94 002EF 49$:    CLRB    VERIFY_PROCESS                              ; 3431
                                    0000V  31 002F1 50$:    BRW     94$
   00000100G  EF 00000000G  EF      D0 002F4 51$:    MOVL    INPUT_VALUE,IDATA+256                       ; 3439
00V00000000G  EF                    00 E1 002FF      BBC     #0,ISAM_ORG,53$                             ; 3445
   00000100   8F 00000100G  EF      D1 00307         CMPL    IDATA+256,#256
                                    00V    1E 00312         BGEQU   55$
00VFFFF8FEE   EF 00000100G  EF      E1 00314         BBC     IDATA+256,C.AOA,55$
                       03 00000108G EF     D1 00320 53$:    CMPL    IDATA+264,#3
                                    03     13 00327         BEQL    .+3
                                    0000V  31 00329         BRW     94$
   00000100   8F 00000100G  EF      D1 0032C         CMPL    IDATA+256,#256
                                    03     1F 00337         BLSSU   .+3
                                    0000V  31 00339         BRW     94$
03 FFFF8FE6   EF 00000100G  EF      E0 0033C         BBS     IDATA+256,C.AOB,.+3
                                    0000V  31 00348         BRW     94$
                                    5C     94 0034B 55$:    CLRB    VERIFY_PROCESS                              ; 3459
                                    0000V  31 0034D         BRW     94$
                          00000000G EF     94 00350 57$:    CLRB    TEST                                        ; 3467
   00000019G  EF 00000000G  EF      90 00356         MOVB    INPUT_VALUE,TEST+25                         ; 3468
   0000001AG  EF 00000000G  EF      D0 00361         MOVL    INPUT_NUMBER,TEST+26                        ; 3469
   00000000G  EF 00000000G  EF      D0 0036C         MOVL    INPUT_NUMBER,DEFAULT_PRINUM                 ; 3470
   00000000G  EF 00000019G  EF      90 00377         MOVB    TEST+25,ACTIVE_PRIMARY                      ; 3471
   00000000G  EF 00000000G  EF      90 00382         MOVB    ACTIVE_PRIMARY,DEFAULT_PRIMARY              ; 3472
                       50 00000000G EF     19 C5 0038D      MULL3   #25,QTAB_OFFSET,R0                          ; 3473
            FFFFFEF2GEF40 00000000G EF              04 28 00395      MOVC3   #4,INPUT_VALUE,QTAB-270[R0]
                       05 00000019G EF     91 003A2         CMPB    TEST+25,#5                                  ; 3475
                                    00V    13 003A9         BEQL    59$
                       0B 00000019G EF     91 003AB         CMPB    TEST+25,#11
                                    03     13 003B2         BEQL    .+3
                                    0000V  31 003B4         BRW     78$
0000001AG  EF       FE   8F          08     00 ED 003B7 59$:    CMPZV   #0,#8,#^XFE,TEST+26                 ; 3483
                                    00V    18 003C1         BGEQ    61$
                                    5C     94 003C3         CLRB    VERIFY_PROCESS                              ; 3485
                                01  8F     9F 003C5 61$:    PUSHAB  #1                                          ; 3487
                          00000000G EF              01 FB 003C8      CALLS   #1,SCAN_DEFINITION
                   00V00000000G EF              00 E1 003CF      BBC     #0,FOUND_AREA,65$                   ; 3489
                       05 00000019G EF     91 003D7         CMPB    TEST+25,#5
                                    00V    12 003DE         BNEQ    65$
                   50 0000001AG EF  00000000G EF     C3 003E0         SUBL3   HIGH_AREA,TEST+26,R0
                       01           50     D1 003EC         CMPL    R0,#1
                                    00V    15 003EF         BLEQ    65$
                                    5C     94 003F1         CLRB    VERIFY_PROCESS                              ; 3497
                   00V00000000G EF              00 E1 003F3 65$:    BBC     #0,FOUND_KEY,69$                    ; 3499
                       0B 00000019G EF     91 003FB         CMPB    TEST+25,#11
                                    00V    12 00402         BNEQ    69$
                   50 0000001AG EF  00000000G EF     C3 00404         SUBL3   HIGH_KEY,TEST+26,R0
                       01           50     D1 00410         CMPL    R0,#1
                                    00V    15 00413         BLEQ    69$
```

```
                                 5C   94 00415          CLRB    VERIFY_PROCESS                          ; 3507
00V00000000G   EF                00   E0 00417 69$:    BBS     #0,FOUND_AREA,73$                       ; 3509
               05 00000019G      EF   91 0041F          CMPB    TEST+25,#5
                                 00V  12 00426          BNEQ    73$
                  0000001AG      EF   D5 00428          TSTL    TEST+26
                                 00V  15 0042E          BLEQ    73$
                                 5C   94 00430          CLRB    VERIFY_PROCESS                          ; 3517
00V00000000G   EF                00   E0 00432 73$:    BBS     #0,FOUND_KEY,79$                        ; 3519
               0B 00000019G      EF   91 0043A          CMPB    TEST+25,#11
                                 00V  12 00441          BNEQ    79$
                  0000001AG      EF   D5 00443          TSTL    TEST+26
                                 00V  15 00449          BLEQ    79$
                                 5C   94 0044B          CLRB    VERIFY_PROCESS                          ; 3527
                                 00V  11 0044D          BRB     79$
                  0000001AG      EF   D4 0044F 78$:     CLRL    TEST+26                                 ; 3533
00V00000000G   EF                00   E0 00455 79$:    BBS     #0,FULL_CHOICE,94$                      ; 3538
   00000000G   EF 00000000G      EF   D0 0045D          MOVL    DEF_HEAD,DEF_CURRENT                    ; 3542
                        00       8F   9F 00468 81$:     PUSHAB  #0                                      ; 3546
                  00000000G      EF   9F 0046B          PUSHAB  TEST
   00000000G   EF                02   FB 00471          CALLS   #2,CURRENT_EQ_TEST
               00V                50   E8 00478          BLBS    R0,83$
   00000000G   EF                00   FB 0047B          CALLS   #0,INCR_CURRENT                         ; 3548
                        00       8F   9F 00482 83$:     PUSHAB  #0
                  00000000G      EF   9F 00485          PUSHAB  TEST
   00000000G   EF                02   FB 0048B          CALLS   #2,CURRENT_EQ_TEST
                                 51   94 00492          CLRB    R1
               52 00000000G      EF   D0 00494          MOVL    DEF_CURRENT,R2
                        01       A2   D5 0049B          TSTL    1(R2)
                                 00V  12 0049E          BNEQ    85$
                                 51   96 004A0          INCB    R1
                  51             50   88 004A2 85$:     BISB2   R0,R1
                  C0             51   E9 004A5          BLBC    R1,81$
                     00000000G   EF   D5 004A8          TSTL    DEF_CURRENT                             ; 3552
                                 00V  13 004AE          BEQL    90$
                           00    8F   9F 004B0          PUSHAB  #0                                      ; 3556
                     00000000G   EF   9F 004B3          PUSHAB  TEST
            00000000G  EF        02   FB 004B9          CALLS   #2,CURRENT_EQ_TEST
                       00V       50   E8 004C0          BLBS    R0,94$
                                 5C   94 004C3          CLRB    VERIFY_PROCESS                          ; 3558
                                 00V  11 004C5          BRB     94$
                                 5C   94 004C7 90$:     CLRB    VERIFY_PROCESS                          ; 3564
                                 00V  11 004C9          BRB     94$
                                         004CB 93$:
                  50             5C   90 004CB 94$:     MOVB    VERIFY_PROCESS,R0                       ; 3576
                                         004CE          RET

; Routine Size: 1231 bytes,    Routine Base: $CODE + 09A22

                                         00000 POST_PROCESS:                                            ; 3621
                                      003C 00000          .WORD   ^M<R2,R3,R4,R5>
                  5E                 04   C2 00002          SUBL2   #4,SP
                  5C                 01   90 00005          MOVB    #1,POST_PROCESS                         ; 3631
      00000042    8F                05 00000000G EF   CF 00008          CASEL   QTAB_OFFSET,#5,#66              ; 3633
                                                0000V    00014          .DISPL  8$
                                                0086     00016          .DISPL  134
                                                0086     00018          .DISPL  134
                                                0086     0001A          .DISPL  134
```

```
0086      0001C         .DISPL  134
0086      0001E         .DISPL  134
0086      00020         .DISPL  134
0086      00022         .DISPL  134
0086      00024         .DISPL  134
0000V     00026         .DISPL  6$
0086      00028         .DISPL  134
0000V     0002A         .DISPL  16$
0000V     0002C         .DISPL  16$
0000V     0002E         .DISPL  16$
0000V     00030         .DISPL  1$
0000V     00032         .DISPL  1$
0000V     00034         .DISPL  1$
0000V     00036         .DISPL  1$
0000V     00038         .DISPL  1$
0000V     0003A         .DISPL  1$
0000V     0003C         .DISPL  1$
0000V     0003E         .DISPL  1$
0000V     00040         .DISPL  1$
0086      00042         .DISPL  134
0086      00044         .DISPL  134
0000V     00046         .DISPL  1$
0000V     00048         .DISPL  20$
0086      0004A         .DISPL  134
0086      0004C         .DISPL  134
0086      0004E         .DISPL  134
0000V     00050         .DISPL  48$
0000V     00052         .DISPL  49$
0086      00054         .DISPL  134
0000V     00056         .DISPL  5$
0000V     00058         .DISPL  4$
0000V     0005A         .DISPL  41$
0000V     0005C         .DISPL  56$
0000V     0005E         .DISPL  4$
0000V     00060         .DISPL  55$
0000V     00062         .DISPL  48$
0000V     00064         .DISPL  49$
0000V     00066         .DISPL  26$
0000V     00068         .DISPL  4$
0086      0006A         .DISPL  134
0000V     0006C         .DISPL  48$
0000V     0006E         .DISPL  49$
0000V     00070         .DISPL  37$
0000V     00072         .DISPL  48$
0000V     00074         .DISPL  49$
0000V     00076         .DISPL  38$
0000V     00078         .DISPL  4$
0000V     0007A         .DISPL  4$
0086      0007C         .DISPL  134
0086      0007E         .DISPL  134
0000V     00080         .DISPL  42$
0000V     00082         .DISPL  5$
0000V     00084         .DISPL  36$
0000V     00086         .DISPL  45$
0000V     00088         .DISPL  10$
0000V     0008A         .DISPL  57$
0000V     0008C         .DISPL  4$
```

```
                                    0000V     0008E          .DISPL  13$
                                    0000V     00090          .DISPL  4$
                                    0000V     00092          .DISPL  48$
                                    0000V     00094          .DISPL  49$
                                    0000V     00096          .DISPL  5$
                                    0000V     00098          .DISPL  61$
                                    0000V  31 0009A          BRW     67$
                                       50  94 0009D 1$:      CLRB    R0                                          ; 3651
                              01 00000000G  EF  D1 0009F      CMPL    INPUT_VALUE,#1
                                    00V     12 000A6         BNEQ    3$
                                       50  96 000A8         INCB    R0
                     00000000G  EF             50  90 000AA 3$:      MOVB    R0,QUERY_FLAG
                                50 00000000G  EF  D0 000B1         MOVL    QTAB_OFFSET,R0                      ; 3652
                     FFFFFFF9GEF40 00000000G  EF  90 000B8         MOVB    QUERY_FLAG,BDATA-7[R0]
                                    0000V     31 000C4         BRW     68$
                                50 00000000G  EF  D0 000C7 4$:      MOVL    QTAB_OFFSET,R0                      ; 3667
                     00000000GEF40 00000000G  EF  D0 000CE         MOVL    INPUT_VALUE,IDATA[R0]
                                    0000V     31 000DA         BRW     68$
                                50 00000000G  EF  D0 000DD 5$:      MOVL    QTAB_OFFSET,R0                      ; 3678
                     00000000GEF40 00000000G  EF  D0 000E4         MOVL    INPUT_VALUE,IDATA[R0]
                           50 00000000G  EF             19  C5 000F0         MULL3   #25,QTAB_OFFSET,R0          ; 3679
                 FFFFFEF2GEF40 00000000G  EF             04  28 000F8         MOVC3   #4,INPUT_VALUE,QTAB-270[R0]
                                    0000V     31 00105         BRW     68$
                     00000000G  EF 00000000G  EF  7D 00108 6$:      MOVQ    NULL_STRING,ANALYSIS_FILENAME_DESC  ; 3687
                                00000000G  EF  9F 00113         PUSHAB  ANALYSIS_FILENAME_DESC              ; 3688
                                00000020G  EF  9F 00119         PUSHAB  SDATA+32
                     00000000G  EF             02  FB 0011F         CALLS   #2,LIB$SCOPY_DXDX
                     00000000G  EF             01  90 00126         MOVB    #1,ANALYSIS_SPECIFIED               ; 3689
                                    0000V     31 0012D         BRW     68$
                     00000000G  EF 00000000G  EF  7D 00130 8$:      MOVQ    NULL_STRING,OUTPUT_FILENAME_DESC    ; 3697
                                00000000G  EF  9F 0013B         PUSHAB  OUTPUT_FILENAME_DESC                ; 3698
                                00000028G  EF  9F 00141         PUSHAB  SDATA+40
                     00000000G  EF             02  FB 00147         CALLS   #2,LIB$SCOPY_DXDX
                                    0000V     31 0014E         BRW     68$
                                50 00000000G  EF  D0 00151 10$:     MOVL    QTAB_OFFSET,R0                      ; 3706
                     00000000GEF40 00000000G  EF  D0 00158         MOVL    INPUT_VALUE,IDATA[R0]
                                       50     94 00164         CLRB    R0                                  ; 3707
                                00000000G  EF  D5 00166         TSTL    INPUT_VALUE
                                    00V     12 0016C         BNEQ    12$
                                       50  96 0016E         INCB    R0
                     00000000G  EF             50  90 00170 12$:     MOVB    R0,FULL_PROMPT
                                    0000V     31 00177         BRW     68$
                                50 00000000G  EF  D0 0017A 13$:     MOVL    QTAB_OFFSET,R0                      ; 3715
                     00000000GEF40 00000000G  EF  D0 00181         MOVL    INPUT_VALUE,IDATA[R0]
                                       50     94 0018D         CLRB    R0                                  ; 3717
                        00000100 8F 00000000G  EF  D1 0018F         CMPL    INPUT_VALUE,#256
                                    00V     1E 0019A         BGEQU   15$
                 00VFFFF8CD7  EF 00000000G  EF  E1 0019C         BBC     INPUT_VALUE,C.AOC,15$
                                       50     96 001A8         INCB    R0
                     00000000G  EF             50  90 001AA 15$:     MOVB    R0,ISAM_ORG
                                    0000V     31 001B1         BRW     68$
                                50 00000000G  EF  D0 001B4 16$:     MOVL    QTAB_OFFSET,R0                      ; 3735
                                50 00000000GEF40  DE 001BB         MOVAL   IDATA[R0],R0
                                51                60  D0 001C3         MOVL    (R0),R1
                                    00V     18 001C6         BGEQ    17$
                                51                51  CE 001C8         MNEGL   R1,R1
           51        63 8F             07             00  ED 001CB 17$:     CMPZV   #0,#7,#^X63,R1
```

```
                                  00V  18 001D1          BGEQ    19$
                                  60   D4 001D3          CLRL    (R0)                                    ; 3737
                           50 00000000G   EF   D0 001D5 19$:     MOVL    QTAB_OFFSET,R0                  ; 3739
                           51 00000000G   EF   D0 001DC          MOVL    QTAB_OFFSET,R1
                FFFFFFE0GEF40 00000000GEF41   4E 001E3          CVTLF   IDATA[R1],RDATA-32[R0]
                           50 00000000G   EF   D0 001F0          MOVL    QTAB_OFFSET,R0                  ; 3740
                           50 FFFFFFE0GEF40   DE 001F7          MOVAL   RDATA-32[R0],R0
                           60 000043C8    8F   46 001FF          DIVF2   #^F100.0,(R0)
                                     0000V  31 00206          BRW     68$
             03 00000000G  EF              00   E1 00209 20$:     BBC     #0,AUTO_TUNE,.+3                ; 3748
                                     0000V  31 00211          BRW     68$
          00V00000000G     EF              00   E1 00214          BBC     #0,REGIS,23$                    ; 3755
                              FFFF8C7D     EF   9F 0021C          PUSHAB  C.AOD                           ; 3757
                                           0B   DD 00222          PUSHL   #11
                              00000000G    EF   9F 00224          PUSHAB  PAS$FV_OUTPUT
                00000000G     EF           03   FB 0022A          CALLS   #3,PAS$WRITE_STRING
                              00000000G    EF   9F 00231          PUSHAB  PAS$FV_OUTPUT
                00000000G     EF           01   FB 00237          CALLS   #1,PAS$WRITELN2
                              00000000G    EF   9F 0023E 23$:     PUSHAB  COL_ONE                         ; 3759
                              00000000G    EF   9F 00244          PUSHAB  LINE_ONE
                00000000G     EF           02   FB 0024A          CALLS   #2,LIB$ERASE_PAGE
                                     0000V  31 00251          BRW     68$
          00V00000000G     EF              00   E0 00254 26$:     BBS     #0,GLOBAL_SET,33$               ; 3773
                              00000000G    EF   D4 0025C          CLRL    PRIMARY_INDEX_BUCKETS           ; 3784
                           50              01   D0 00262          MOVL    #1,R0                           ; 3786
                           51              50   D0 00265 28$:     MOVL    R0,I
                           52 00000000GEF41   DE 00268          MOVAL   INIT_PRIMARY_BUCKETS[I],R2      ; 3790
                00000200   8F                62   D1 00270          CMPL    (R2),#512
                                              00V  15 00277          BLEQ    30$
                           62        0200     8F   3C 00279          MOVZWL  #512,(R2)                       ; 3792
                           53 00000000GEF41   DE 0027E 30$:     MOVAL   ADDED_PRIMARY_BUCKETS[I],R3     ; 3794
                00000200   8F                63   D1 00286          CMPL    (R3),#512
                                              00V  15 0028D          BLEQ    32$
                           63        0200     8F   3C 0028F          MOVZWL  #512,(R3)                       ; 3796
                     52 00000000G  EF          62   C1 00294 32$:     ADDL3   (R2),PRIMARY_INDEX_BUCKETS,R2   ; 3798
00000000G            EF                       52   C1 0029C          ADDL3   R2,(R3),PRIMARY_INDEX_BUCKETS
                     BD                       50   F3 002A4          AOBLEQ  #31,R0,28$
                                              50 00000000G   EF   D0 002A8          MOVL    PRIMARY_INDEX_BUCKETS,R0        ; 3808
                                              51 000000F0G   EF   D0 002AF          MOVL    IDATA+240,R1
                                 000000B8G    EF        04 A140   9E 002B6          MOVAB   4(R1)[R0],IDATA+184
                                 00007FFF     8F 000000B8G   EF   D1 002BF 33$:     CMPL    IDATA+184,#32767        ; 3816
                                                             03   14 002CA          BGTR    .+3
                                                       0000V  31 002CC          BRW     68$
                                 000000B8G    EF             7FFF 8F   3C 002CF          MOVZWL  #32767,IDATA+184        ; 3818
                                                       0000V  31 002D8          BRW     68$
                                              51 00000000G   EF   D0 002DB 36$:     MOVL    QTAB_OFFSET,R1          ; 3824
                                 000000C0G    EF 00000000GEF41   D0 002E2          MOVL    IDATA[R1],IDATA+192
                                                       0000V  31 002EE          BRW     68$
                                              51 00000000G   EF   D0 002F1 37$:     MOVL    SEGMENT_NUMBER,R1       ; 3828
                                              50 00000000G   EF   D0 002F8          MOVL    QTAB_OFFSET,R0
                                 00000000GEF41 00000000GEF40   D0 002FF          MOVL    IDATA[R0],SEGMENT_POSITION[R1]
                                                       0000V  31 0030C          BRW     68$
                                              51 00000000G   EF   D0 0030F 38$:     MOVL    SEGMENT_NUMBER,R1       ; 3834
                                                             50   94 00316          CLRB    R0
                                                 000000D8G   EF   D5 00318          TSTL    IDATA+216
                                                             00V  15 0031E          BLEQ    40$
                                                             50   96 00320          INCB    R0
```

```
                    00000000GEF41            50    90 00322 40$:    MOVB    R0,SEGMENT_WANTED[R1]
                                 51 00000000G  EF    D0 00327         MOVL    SEGMENT_NUMBER,R1                    ; 3835
                                 50 00000000G  EF    D0 00331         MOVL    QTAB_OFFSET,R0
                    00000000GEF41 00000000GEF40    D0 00338         MOVL    IDATA[R0],SEGMENT_LENGTH[R1]
                                           0000V   31 00345         BRW     68$
                                 51 00000000G  EF    D0 00348 41$:    MOVL    QTAB_OFFSET,R1                       ; 3841
                    00000000G  EF 00000000GEF41    C2 0034F         SUBL2   IDATA[R1],CUR_MAX_REC
                                           0000V   31 0035B         BRW     68$
                                 51 00000000G  EF    D0 0035E 42$:    MOVL    QTAB_OFFSET,R1                       ; 3845
                                 51 00000000GEF41    DE 00365         MOVAL   IDATA[R1],R1
                                              61    D5 0036D         TSTL    (R1)
                                              03    19 0036F         BLSS    .+3
                                           0000V   31 00371         BRW     68$
                                              61    D4 00374         CLRL    (R1)                                 ; 3847
                                           0000V   31 00376         BRW     68$
                                 03 000000F8G  EF    D1 00379 45$:    CMPL    IDATA+248,#3                         ; 3853
                                              03    19 00380         BLSS    .+3
                                           0000V   31 00382         BRW     68$
                                    00000024G  EF    D4 00385         CLRF    RDATA+36                             ; 3857
                                    00000020G  EF    D4 0038B         CLRF    RDATA+32                             ; 3858
                                    00000028G  EF    D4 00391         CLRF    RDATA+40                             ; 3859
                                           0000V   31 00397         BRW     68$
                                 51 00000000G  EF    D0 0039A 48$:    MOVL    QTAB_OFFSET,R1                       ; 3873
                       00000014G  EF 00000000GEF41    D0 003A1         MOVL    IDATA[R1],IDATA+20
                    51 00000000G  EF              19    C5 003AD         MULL3   #25,QTAB_OFFSET,R1                   ; 3874
                                 50 00000000G  EF    D0 003B5         MOVL    QTAB_OFFSET,R0
                          FC     AD 00000000GEF40    DE 003BC         MOVAL   IDATA[R0],-4(FP)
                FFFFFF0FGEF41     FC    BD              04    28 003C5         MOVC3   #4,a-4(FP),QTAB-241[R1]
                                           0000V   31 003CF         BRW     68$
                                 50 00000000G  EF    D0 003D2 49$:    MOVL    QTAB_OFFSET,R0                       ; 3886
                       00000010G  EF 00000000GEF40    D0 003D9         MOVL    IDATA[R0],IDATA+16
                 21                24 00000000G  EF    CF 003E5         CASEL   QTAB_OFFSET,#36,#33                  ; 3888
                                           0000V       003ED         .DISPL  52$
                                           0044        003EF         .DISPL  68
                                           0044        003F1         .DISPL  68
                                           0044        003F3         .DISPL  68
                                           0044        003F5         .DISPL  68
                                           0044        003F7         .DISPL  68
                                           0044        003F9         .DISPL  68
                                           0044        003FB         .DISPL  68
                                           0044        003FD         .DISPL  68
                                           0000V       003FF         .DISPL  50$
                                           0044        00401         .DISPL  68
                                           0044        00403         .DISPL  68
                                           0044        00405         .DISPL  68
                                           0044        00407         .DISPL  68
                                           0000V       00409         .DISPL  52$
                                           0044        0040B         .DISPL  68
                                           0044        0040D         .DISPL  68
                                           0000V       0040F         .DISPL  52$
                                           0044        00411         .DISPL  68
                                           0044        00413         .DISPL  68
                                           0044        00415         .DISPL  68
                                           0044        00417         .DISPL  68
                                           0044        00419         .DISPL  68
                                           0044        0041B         .DISPL  68
                                           0044        0041D         .DISPL  68
```

```
                                0044      0041F          .DISPL  68
                                0044      00421          .DISPL  68
                                0044      00423          .DISPL  68
                                0044      00425          .DISPL  68
                                0044      00427          .DISPL  68
                                0044      00429          .DISPL  68
                                0044      0042B          .DISPL  68
                                0044      0042D          .DISPL  68
                                0000V     0042F          .DISPL  51$
                                   00V 11 00431          BRB     53$
                        00000064   8F    DF 00433 50$:    PUSHAL  #100                                    ; 3890
                        00000032   8F    DF 00439         PUSHAL  #50
                7FC1    CF               02 0043F         CALLS   #2,SPREAD_LOW_HIGH
                                   00V 11 00444          BRB     54$
                        00000000G  EF    9F 00446 51$:    PUSHAB  CUR_MAX_REC                             ; 3892
                        00000001   8F    DF 0044C         PUSHAL  #1
                7FC1    CF               02 00452         CALLS   #2,SPREAD_LOW_HIGH
                                   00V 11 00457          BRB     54$
                        7FFFFFFE   8F    DF 00459 52$:    PUSHAL  #2147483646                             ; 3896
                        00000001   8F    DF 0045F         PUSHAL  #1
                7FC1    CF               02 00465         CALLS   #2,SPREAD_LOW_HIGH
                                   00V 11 0046A          BRB     54$
                                          0046C 53$:
                                   00V 11 0046C 54$:     BRB     68$
00000000G       EF 000000ACG    EF    D0 0046E 55$:     MOVL    IDATA+172,IDATA                         ; 3908
                                   00V 11 00479          BRB     68$
                   50 00000000G EF    D0 0047B 56$:     MOVL    QTAB_OFFSET,R0                          ; 3914
00000000GEF40 00000000G         EF    D0 00482          MOVL    INPUT_VALUE,IDATA[R0]
00000108G       EF              07    D0 0048E          MOVL    #7,IDATA+264                            ; 3919
                        00000000G  EF    94 00495         CLRB    MAIN_LEVEL                              ; 3924
                                   00V 11 0049B          BRB     68$
                                   50    94 0049D 57$:    CLRB    R0                                      ; 3939
                   0D 00000100G    EF    D1 0049F         CMPL    IDATA+256,#13
                                   00V 13 004A6          BEQL    60$
                   0C 00000100G    EF    D1 004A8         CMPL    IDATA+256,#12
                                   00V 13 004AF          BEQL    60$
                                   50    96 004B1         INCB    R0
00000000G       EF              50    90 004B3 60$:     MOVB    R0,VARIABLE_RECORDS
                                   00V 11 004BA          BRB     68$
                   05 00000000G    EF    91 004BC 61$:     CMPB    ACTIVE_PRIMARY,#5                       ; 3951
                                   00V 12 004C3          BNEQ    63$
00000000G       EF 00000000G    EF    D0 004C5          MOVL    INPUT_NUMBER,ACTIVE_AREA                ; 3953
                                   00V 11 004D0          BRB     68$
                   0B 00000000G    EF    91 004D2 63$:     CMPB    ACTIVE_PRIMARY,#11                      ; 3955
                                   00V 12 004D9          BNEQ    68$
00000084G       EF 00000000G    EF    D0 004DB          MOVL    INPUT_NUMBER,IDATA+132                  ; 3957
                                   00V 11 004E6          BRB     68$
                                          004E8 67$:
                   50              5C    90 004E8 68$:     MOVB    POST_PROCESS,R0                         ; 3967
                                         04 004EB          RET

; Routine Size: 1260 bytes,    Routine Base: $CODE + 09EF1

                                      0000 00000         .ENTRY  QUERY,^M<>                              ; 4016
                   5C       04    BC    D0 00002         MOVL    @4(R12),OFFSET
00000000G       EF              5C    D0 00006         MOVL    OFFSET,QTAB_OFFSET                      ; 4260
                83FC    CF              00    FB 0000D         CALLS   #0,PRE_PROCESS                  ; 4262
```

```
                                 00V            50   E9 00012          BLBC    R0,4$
                              00V  AF           00   FB 00015 2$:      CALLS   #0,THE_QUESTION                         ; 4271
                       F4 00000000G  EF         00   E0 00019          BBS     #0,SYS$INPUT_ERROR,2$
                                        00000000G  EF   9F 00021          PUSHAB  INPUT_DESC                              ; 4275
                          00000000G  EF           01   FB 00027          CALLS   #1,STR$FREE1_DX
                              9EF1   CF           00   FB 0002E 4$:      CALLS   #0,POST_PROCESS                         ; 4282
                                 00V             50   E9 00033          BLBC    R0,6$
                                 50 00000000G    EF   D0 00036          MOVL    QTAB_OFFSET,R0                          ; 4284
                          FFFFFFF5GEF40          01   90 0003D          MOVB    #1,VDATA-11[R0]
                                 50 00000000G    EF   90 00045 6$:      MOVB    QUERY_FLAG,QUERY                        ; 4289
                                                 04   90 0004C          RET                                             ; 4291

; Routine Size: 77 bytes,    Routine Base: $CODE + 0A3DD

                                                      00000 THE_QUESTION:                                              ; 4018
                                                 000C 00000          .WORD   ^M<R2,R3>
                                    5E              10   C2 00002          SUBL2   #16,SP
                                              F8    AD   D4 00005          CLRL    -8(FP)
                                    6D 00000000G    EF   9E 00008          MOVAB   PAS$HANDLER,(FP)
                                    29 00000000G    EF   D1 0000F          CMPL    QTAB_OFFSET,#41                         ; 4025
                                                    00V  12 00016          BNEQ    2$
                          00000000G  EF             01   90 00018          MOVB    #1,MAIN_LEVEL                           ; 4029
                                        00000000G   EF   94 0001F          CLRB    MAIN_CTRLZ                              ; 4030
                                        00000000G   EF   94 00025          CLRB    CONTROL_ZEE_TYPED                       ; 4031
                                        00000000G   EF   94 0002B 2$:      CLRB    SYS$INPUT_ERROR                         ; 4038
                                 F8     AD 00000000G   EF   9E 00031          MOVAB   SYS$INPUT_COND_HANDLER,FP-8             ; 4039
                       00V00000000G     EF             00   EC 00039          BBS     #0,AUTO_TUNE,6$                         ; 4041
                       00V00000000G     EF             00   E1 00041          BBC     #0,TEMP_FULL_PROMPT,5$                  ; 4045
                              3E9F      CF             00   FB 00049          CALLS   #0,WRITE_HELP                           ; 4047
                              49C2      CF             00   FB 0004E 5$:      CALLS   #0,WRITE_QUESTION                       ; 4049
                          5C 00000000G  EF             19   C5 00053 6$:      MULL3   #25,QTAB_OFFSET,R12                     ; 4053
                 50 FFFFFEEDGEF4C                    20   00   EE 0005B          EXTV    #0,#32,QTAB-275[R12],R0
                       06                            00   50   CF 00065          CASEL   R0,#0,#6
                                                          0000V    00069          .DISPL  7$
                                                          0000V    0006B          .DISPL  26$
                                                          0000V    0006D          .DISPL  46$
                                                          0000V    0006F          .DISPL  26$
                                                          0000V    00071          .DISPL  46$
                                                          0000V    00073          .DISPL  47$
                                                          0000V    00075          .DISPL  56$
                                                          0000V 31 00077          BRW     57$
                                    50 00000000G    EF   D0 0007A 7$:      MOVL    QTAB_OFFSET,R0                          ; 4059
                          FFFFFFB0GEF40 00000000G   EF   7D 00081          MOVQ    NULL_STRING,SDATA-80[R0]
                       00V00000000G     EF             00   E1 0008D          BBC     #0,TAKE_DEFAULTS,9$                     ; 4061
                                        00000104G   EF   D5 00095          TSTL    IDATA+260
                                                    00V  13 00099          BEQL    10$
                       00V00000000G     EF             00   E1 0009D 9$:      BBC     #0,AUTO_TUNE,13$
                       03 00000000G     EF             00   E1 000A5 10$:     BBC     #0,AUTO_TUNE,.+3                        ; 4073
                                                    0000V 31 000AD          BRW     18$
                                        33334033    8F   DF 000B0          PUSHAF  #^F0.7                                  ; 4075
                          00000000G     EF             01   FB 000B6          CALLS   #1,LIB$WAIT
                                                    0000V 31 000BD          BRW     18$
                       00V00000000G     EF             30   E0 000C0 13$:     BBS     #48,PAS$FV_INPUT,14$                    ; 4083
                                        00000000G   EF   9F 000C8          PUSHAB  PAS$FV_INPUT
                          00000000G     EF             01   FB 000CE          CALLS   #1,PAS$LOOK_AHEAD
                       00V00000000G     EF             31   E0 000D5 14$:     BBS     #49,PAS$FV_INPUT,16$
                                        00000000G   EF   9F 000DD          PUSHAB  PAS$FV_INPUT                            ; 4087
```

```
                00000000G  EF                01   FB 000E3          CALLS   #1,PAS$RESET2
                                              00   DD 000EA          PUSHL   #0                                      ; 4088
                                              00   DD 000EC          PUSHL   #0
                                              00   DD 000EE          PUSHL   #0
                           00B3804B           8F   DD 000F0          PUSHL   #11763787
                00000000G  EF                04   FB 000F6          CALLS   #4,LIB$SIGNAL
                           000000FF           8F   DD 000FD 16$:     PUSHL   #255                                    ; 4092
                           00000000G          EF   9F 00103          PUSHAB  PAS$FV_INPUT
                           00000000G          EF   9F 00109          PUSHAB  TEMP_STRING255
                00000000G  EF                03   FB 0010F          CALLS   #3,PAS$READ_STRING
                           00000000G          EF   9F 00116          PUSHAB  PAS$FV_INPUT
                00000000G  EF                01   FB 0011C          CALLS   #1,PAS$READLN2
                      F0   AD 010E00FF        8F   D0 00123          MOVL    #17694975,-16(FP)                       ; 4093
                      F4   AD 00000000G       EF   9E 0012B          MOVAB   TEMP_STRING255,-12(FP)
                                 F0           AD   9F 00133          PUSHAB  -16(FP)
                           50 00000000G       EF   D0 00136          MOVL    QTAB_OFFSET,R0
                              FFFFFFB0GEF40        7F 0013D          PUSHAQ  SDATA-80[R0]
                00000000G  EF                02   FB 00144          CALLS   #2,STR$TRIM
                           00000000G          EF   9F 0014B          PUSHAB  INPUT_DESC                              ; 4094
                           50 00000000G       EF   D0 00151          MOVL    QTAB_OFFSET,R0
                              FFFFFFB0GEF40        7F 00158          PUSHAQ  SDATA-80[R0]
                00000000G  EF                02   FB 0015F          CALLS   #2,LIB$SCOPY_DXDX
                00000014G  EF  00000004G     EF   D0 00166          MOVL    INPUT_DESC+4,PARAM_BLOCK+20             ; 4095
                00000010G  EF  00000000G     EF   3C 00171          MOVZWL  INPUT_DESC,PARAM_BLOCK+16               ; 4096
           00V00000000G  EF                00   E0 0017C 18$:     BBS     #0,AUTO_TUNE,20$                        ; 4100
                           00000000G          EF   9F 00184          PUSHAB  CRLF                                    ; 4102
                                              02   DD 0018A          PUSHL   #2
                           00000000G          EF   9F 0018C          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                03   FB 00192          CALLS   #3,PAS$WRITE_STRING
                           00000000G          EF   9F 00199          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                01   FB 0019F          CALLS   #1,PAS$WRITELN2
           03   00000000G  EF                00   E0 001A6 20$:     BBS     #0,JOURNAL_ENABLED,.+3                  ; 4108
                                       0000V  31   001AE          BRW     58$
                           50 00000000G       EF   D0 001B1          MOVL    QTAB_OFFSET,R0                          ; 4110
                           50 FFFFFFB0GEF40        7E 001B8          MOVAQ   SDATA-80[R0],R0
                                              60   B5 001C0          TSTW    (R0)
                                              00V  1B 001C2          BLEQU   23$
                              7E              60   3C 001C4          MOVZWL  (R0),-(SP)                              ; 4112
                                              00   DD 001C7          PUSHL   #0
                           51 00000000G       EF   D0 001C9          MOVL    QTAB_OFFSET,R1
                           51 FFFFFFB4GEF41        7E 001D0          MOVAQ   SDATA-76[R1],R1
                                    00        B1   9F 001D8          PUSHAB  @0(R1)
                              000000FF        8F   DD 001DB          PUSHL   #255
                              00000000G       EF   9F 001E1          PUSHAB  JOURNAL_FILE
                00000000G  EF                05   FB 001E7          CALLS   #5,PAS$WRITE_STRING
                           00000000G          EF   9F 001EE          PUSHAB  JOURNAL_FILE
                00000000G  EF                01   FB 001F4          CALLS   #1,PAS$WRITELN2
                                       0000V  31   001FB          BRW     58$
                           00000000G          EF   9F 001FE 23$:     PUSHAB  JOURNAL_FILE                            ; 4120
                00000000G  EF                01   FB 00204          CALLS   #1,PAS$WRITELN2
                                       0000V  31   0020B          BRW     58$
                           FFFFFEF2GEF4C       9F   0020E 26$:     PUSHAB  QTAB-270[R12]                           ; 4129
                           FFFFFEF1GEF4C       9F   00215          PUSHAB  QTAB-271[R12]
                           50 00000000G       EF   D0 0021C          MOVL    QTAB_OFFSET,R0
                              00000000GEF40        DF 00223          PUSHAL  IDATA[R0]
                00000000G  EF                03   FB 0022A          CALLS   #3,NUMBER_INPUT
                                              50   94 00231          CLRB    R0                                      ; 4138
```

```
                    51 00000000G   EF    D0 00233          MOVL    QTAB_OFFSET,R1
                    51 00000000GEF41    D0 0023A          MOVL    IDATA[R1],R1
                               00V  13 00242          BEQL    28$
                                50  96 00244          INCB    R0
                                52  94 00246 28$:     CLRB    R2
                    39 00000000G   EF    D1 00248          CMPL    QTAB_OFFSET,#57
                               00V  12 0024F          BNEQ    31$
                    03 00000108G   EF    D1 00251          CMPL    IDATA+264,#3
                               00V  13 00258          BEQL    31$
                                52  96 0025A          INCB    R2
                    52              50  8A 0025C 31$:     BICB2   R0,R2
                                50  94 0025F          CLRB    R0
                00V00000013G   EF       00  E1 00261          BBC     #0,BDATA+19,36$
                                   00000000G   EF    D5 00269          TSTL    SEGMENT_NUMBER
                               00V  13 0026F          BEQL    36$
                    36 00000000G   EF    D1 00271          CMPL    QTAB_OFFSET,#54
                               00V  12 00278          BNEQ    36$
                                51  D5 0027A          TSTL    R1
                               00V  12 0027C          BNEQ    36$
                                50  96 0027E          INCB    R0
                    50              52  88 00280 36$:     BISB2   R2,R0
                    00V             50  E8 00283          BLBS    R0,45$
                                50  94 00286          CLRB    R0                                      ; 4168
            52 00000000G   EF       19  C5 00288          MULL3   #25,QTAB_OFFSET,R2
51 FFFFFEF6GEF42           20       00  EC 00290          CMPV    #0,#32,QTAB-266[R2],R1
                               00V  15 0029A          BLEQ    40$
                                50  96 0029C          INCB    R0
                                53  94 0029E 40$:     CLRB    R3
51 FFFFFEFAGEF42           20       00  EC 002A0          CMPV    #0,#32,QTAB-262[R2],R1
                               00V  18 002AA          BGEQ    42$
                                53  96 002AC          INCB    R3
                    53              50  88 002AE 42$:     BISB2   R0,R3
                    00V             53  E9 002B1          BLBC    R3,45$
                                00  DD 002B4          PUSHL   #0                                      ; 4174
                                00  DD 002B6          PUSHL   #0
                                00  DD 002B8          PUSHL   #0
                       00B38038    8F    DD 002BA          PUSHL   #11763768
            00000000G   EF             04  FB 002C0          CALLS   #4,LIB$SIGNAL
                                0000V  31 002C7 45$:     BRW     58$
                       FFFFFEF2GEF4C    9F 002CA 46$:     PUSHAB  QTAB-270[R12]                       ; 4185
                       FFFFFEF1GEF4C    9F 002D1          PUSHAB  QTAB-271[R12]
                       FFFFFF02GEF4C    9F 002D8          PUSHAB  QTAB-254[R12]
                       FFFFFEFEGEF4C    9F 002DF          PUSHAB  QTAB-258[R12]
            00000000G   EF             04  FB 002E6          CALLS   #4,PARSE_INPUT
                               00V  11 002ED          BRB     58$
        00V00000000G   EF             00  E1 002EF 47$:     BBS     #0,AUTO_TUNE,58$                    ; 4200
        00V00000000G   EF             30  E0 002F7          BBS     #48,PAS$FV_INPUT,50$                ; 4212
                       00000000G   EF    9F 002FF          PUSHAB  PAS$FV_INPUT
            00000000G   EF             01  FB 00305          CALLS   #1,PAS$LOOK_AHEAD
        00V00000000G   EF             31  E0 0030C 50$:     BBS     #49,PAS$FV_INPUT,52$
                       00000000G   EF    9F 00314          PUSHAB  PAS$FV_INPUT                        ; 4216
            00000000G   EF             01  FB 0031A          CALLS   #1,PAS$RESET2
                                00  DD 00321          PUSHL   #0                                      ; 4217
                                00  DD 00323          PUSHL   #0
                                00  DD 00325          PUSHL   #0
                       00B3804B    8F    DD 00327          PUSHL   #11763787
            00000000G   EF             04  FB 0032D          CALLS   #4,LIB$SIGNAL
```

```
                                00000000G   EF    9F 00334 52$:     PUSHAB  PAS$FV_INPUT                          ; 4221
                        00000000G   EF          01   FB 0033A           CALLS   #1,PAS$READLN2
                    00V00000000G    EF          00   E1 00341           BBC     #0,JOURNAL_ENABLED,58$           ; 4223
                                00000000G   EF    9F 00349           PUSHAB  JOURNAL_FILE                     ; 4225
                        00000000G   EF          01   FB 0034F           CALLS   #1,PAS$WRITELN2
                                            00V  11   00356           BRB     59$
                                            00V  11   00358 56$:      BRB     58$
                                                      0035A 57$:
                            9A22    CF          00   FB 0035A 58$:     CALLS   #0,VERIFY_PROCESS                ; 4244
                                    00V         50   E8 0035F           BLBS    R0,60$
                                                00   DD 00362           PUSHL   #0                               ; 4246
                                                00   DD 00364           PUSHL   #0
                                                00   DD 00366           PUSHL   #0
                                00B38038    8F      DD 00368           PUSHL   #11763768
                        00000000G   EF          04   FB 0036E           CALLS   #4,LIB$SIGNAL
                                00000000G   EF    94 00375 60$:     CLRB    TEMP_FULL_PROMPT                 ; 4251
                                                04   0037B           RET                                      ; 4253

; Routine Size: 892 bytes,    Routine Base: $CODE + 0A42A

                                                      00000 ASK_KEY_DUPS:                                    ; 4340
                                                0000  00000           .WORD   ^M<>
                                0000001E    8F      DF 00002           PUSHAL  #30                              ; 4344
                            A3DD    CF          01   FB 00008           CALLS   #1,QUERY
                                    00V         50   E9 0000D           BLBC    R0,3$
                                0000003B    8F      DF 00010           PUSHAL  #59                              ; 4346
                            A3DD    CF          01   FB 00016           CALLS   #1,QUERY
                                            00V  11   0001B           BRB     4$
                                000000ECG   EF    D4 0001D 3$:      CLRL    IDATA+236                        ; 4350
                                                04   00023 4$:      RET                                      ; 4352

; Routine Size: 36 bytes,    Routine Base: $CODE + 0A7A6

                                                      00000 ASK_GLOBAL_WANTED:                               ; 4401
                                                0000  00000           .WORD   ^M<>
                                0000001B    8F      DF 00002           PUSHAL  #27                              ; 4408
                            A3DD    CF          01   FB 00008           CALLS   #1,QUERY
                                    00V         50   E9 0000D           BLBC    R0,3$
                                0000002E    8F      DF 00010           PUSHAL  #46                              ; 4410
                            A3DD    CF          01   FB 00016           CALLS   #1,QUERY
                                            00V  11   0001B           BRB     4$
                                000000B8G   EF    D4 0001D 3$:      CLRL    IDATA+184                        ; 4414
                                                04   00023 4$:      RET                                      ; 4416

; Routine Size: 36 bytes,    Routine Base: $CODE + 0A7CA

                                                      00000 ASK_KEY_COMP:                                    ; 4465
                                                0000  00000           .WORD   ^M<>
                                00000013    8F      DF 00002           PUSHAL  #19                              ; 4472
                            A3DD    CF          01   FB 00008           CALLS   #1,QUERY
                                    00V         50   E9 0000D           BLBC    R0,3$
                                00000010    8F      DF 00010           PUSHAL  #16                              ; 4474
                            A3DD    CF          01   FB 00016           CALLS   #1,QUERY
                                            00V  11   0001B           BRB     4$
                                00000020G   EF    D4 0001D 3$:      CLRF    RDATA+32                         ; 4478
                                                04   00023 4$:      RET                                      ; 4480
```

```
; Routine Size: 36 bytes,    Routine Base: $CODE + 0A7EE

                                        00000 ASK_REC_COMP:                                     ; 4529
                                  0000  00000         .WORD   ^M<>
                    00000014    8F    DF  00002         PUSHAL  #20                           ; 4536
              A3DD  CF          01    FB  00008         CALLS   #1,QUERY
                    00V         50    E9  0000D         BLBC    R0,3$
                    00000011    8F    DF  00010         PUSHAL  #17                           ; 4538
              A3DD  CF          01    FB  00016         CALLS   #1,QUERY
                                00V   11  0001B         BRB     4$
                    00000024G   EF    D4  0001D 3$:     CLRF    RDATA+36                      ; 4542
                                      04  00023 4$:     RET                                   ; 4544

; Routine Size: 36 bytes,    Routine Base: $CODE + 0A812

                                        00000 ASK_IDX_COMP:                                     ; 4593
                                  0000  00000         .WORD   ^M<>
                    00000015    8F    DF  00002         PUSHAL  #21                           ; 4600
              A3DD  CF          01    FB  00008         CALLS   #1,QUERY
                    00V         50    E9  0000D         BLBC    R0,3$
                    00000012    8F    DF  00010         PUSHAL  #18                           ; 4602
              A3DD  CF          01    FB  00016         CALLS   #1,QUERY
                                00V   11  0001B         BRB     4$
                    00000028G   EF    D4  0001D 3$:     CLRF    RDATA+40                      ; 4606
                                      04  00023 4$:     RET                                   ; 4608

; Routine Size: 36 bytes,    Routine Base: $CODE + 0A836

                                        00000 ASK_MEAN_RECORD_SIZE:                             ; 4655
                                  0000  00000         .WORD   ^M<>
00V00000000G  EF                  00    E1  00002         BBC     #0,ISAM_ORG,4$              ; 4663
00V00000000G  EF                  00    E1  0000A         BBC     #0,ISAM_ORG,3$
                    00000084G   EF    D5  00012         TSTL    IDATA+132
                                00V   13  00018         BEQL    4$
00V0000002FG  EF                  00    E0  0001A 3$:     BBS     #0,VDATA+47,13$
                    0000003A    8F    DF  00022 4$:     PUSHAL  #58                           ; 4676
        A3DD  CF                  01    FB  00028         CALLS   #1,QUERY
                    0F  00000100G EF    D1  0002D         CMPL    IDATA+256,#15               ; 4685
                                00V   12  00034         BNEQ    8$
                    00000028    8F    DF  00036         PUSHAL  #40                           ; 4687
        A3DD  CF                  01    FB  0003C         CALLS   #1,QUERY
00V00000000G  EF                  00    E1  00041 8$:     BBC     #0,VARIABLE_RECORDS,11$     ; 4689
                    00000039    8F    DF  00049         PUSHAL  #57                           ; 4691
        A3DD  CF                  01    FB  0004F         CALLS   #1,QUERY
                                00V   11  00054         BRB     13$
   000000E4G  EF 000000E8G  EF    D0  00056 11$:    MOVL    IDATA+232,IDATA+228               ; 4695
                                      04  00061 13$:    RET                                   ; 4699

; Routine Size: 98 bytes,    Routine Base: $CODE + 0A85A

                                        00000 ASK_KEY_SIZE:                                     ; 4747
                                  0000  00000         .WORD   ^M<>
00V000000013G  EF                 00    E1  00002         BBC     #0,BDATA+19,10$             ; 4754
                    00000000G   EF    D4  0000A         CLRL    SEGMENT_NUMBER                ; 4758
                    00000036    8F    DF  00010 2$:     PUSHAL  #54                           ; 4762
        A3DD  CF                  01    FB  00016         CALLS   #1,QUERY
                    00000000G   EF    D6  0001B         INCL    SEGMENT_NUMBER                ; 4764
```

```
                    000000D8G  EF   D5 00021          TSTL    IDATA+216
                               00V  13 00027          BEQL    5$
                            07 00000000G  EF   D1 00029          CMPL    SEGMENT_NUMBER,#7
                               DE   15 00030          BLEQ    2$
                    000000D8G  EF   D5 00032 5$:      TSTL    IDATA+216                          ; 4768
                               00V  12 00038          BNEQ    9$
                            50 00000000G  EF   D0 0003A          MOVL    SEGMENT_NUMBER,R0          ; 4772
                            07            50   D1 00041          CMPL    R0,#7
                               00V  14 00044          BGTR    9$
                            51            50   D0 00046 7$:      MOVL    R0,I
                    00000000GEF41   94 00049          CLRB    SEGMENT_WANTED[I]                  ; 4774
                F2          50            07   F3 00050          AOBLEQ  #7,R0,7$
                   000000D8G  EF 00000000G  EF   D0 00054 9$:      MOVL    SEGMENT_LENGTH,IDATA+216   ; 4778
                               00V  11 0005F          BRB     12$
                        00000036  8F   DF 00061 10$:     PUSHAL  #54                        ; 4784
                      A3DD   CF            01   FB 00067          CALLS   #1,QUERY
                                           04 0006C 12$:     RET                                ; 4786

; Routine Size: 109 bytes,    Routine Base: $CODE + 0A8BC

                                        00000 ASK_KEY_POSITION:                                 ; 4834
                                   0000 00000          .WORD   ^M<>
                00V00000013G  EF            00   E1 00002          BBC     #0,BDATA+19,6$     ; 4838
                                            5C   D4 0000A          CLRL    R12                ; 4842
                   00000000G  EF            5C   D0 0000C 2$:      MOVL    R12,SEGMENT_NUMBER
                            50 00000000G  EF   D0 00013          MOVL    SEGMENT_NUMBER,R0          ; 4846
                00V00000000GEF40            00   E1 0001A          BBC     #0,SEGMENT_WANTED[R0],5$
                        00000033  8F   DF 00023          PUSHAL  #51                        ; 4848
                      A3DD   CF            01   FB 00029          CALLS   #1,QUERY
                DA          5C            07   F3 0002E 5$:      AOBLEQ  #7,R12,2$
                   000000CCG  EF 00000000G  EF   D0 00032          MOVL    SEGMENT_POSITION,IDATA+204 ; 4852
                               00V  11 0003D          BRB     8$
                        00000033  8F   DF 0003F 6$:      PUSHAL  #51                        ; 4858
                      A3DD   CF            01   FB 00045          CALLS   #1,QUERY
                                           04 0004A 8$:      RET                                ; 4860

; Routine Size: 75 bytes,    Routine Base: $CODE + 0A929

                                        00000 ASK_TEST_SECONDARY:                               ; 4909
                                   0000 00000          .WORD   ^M<>
                      00V  AF            00   FB 00002 1$:      CALLS   #0,THE_QUESTION            ; 6003
                F4 00000000G  EF            00   E0 00006          BBS     #0,SYS$INPUT_ERROR,1$
                            00000000G  EF   9F 0000E          PUSHAB  INPUT_DESC                 ; 6007
                   00000000G  EF            01   FB 00014          CALLS   #1,STR$FREE1_DX
                                           04 0001B          RET                                ; 6009

; Routine Size: 28 bytes,    Routine Base: $CODE + 0A974

                                        00000 THE_QUESTION:                                     ; 4911
                                   0000 00000          .WORD   ^M<>
                         5E            10   C2 00002          SUBL2   #16,SP
                               F8      AD   D4 00005          CLRL    -8(FP)
                         6D 00000000G  EF   9E 00008          MOVAB   PAS$HANDLER,(FP)
                            00000000G  EF   D4 0000F          CLRL    EDF$GL_SECNUM              ; 4918
                            00000000G  EF   94 00015          CLRB    SYS$INPUT_ERROR            ; 4919
                      F8    AD 00000000G  EF   9E 0001B          MOVAB   SYS$INPUT_COND_HANDLER,FP-8 ; 4920
                            50 00000000G  EF   9A 00023          MOVZBL  ACTIVE_PRIMARY,R0          ; 4922
```

```
                    0E      01          50   8F 0002A          CASEB   R0,#1,#14
                                  0000V      0002E          .DISPL  3$
                                  001E       00030          .DISPL  30
                                  0000V      00032          .DISPL  15$
                                  0000V      00034          .DISPL  16$
                                  0000V      00036          .DISPL  17$
                                  0000V      00038          .DISPL  29$
                                  0000V      0003A          .DISPL  41$
                                  0000V      0003C          .DISPL  53$
                                  0000V      0003E          .DISPL  1$
                                  0000V      00040          .DISPL  65$
                                  0000V      00042          .DISPL  77$
                                  0000V      00044          .DISPL  89$
                                  0000V      00046          .DISPL  101$
                                  0000V      00048          .DISPL  113$
                                  0000V      0004A          .DISPL  2$
                                  0000V   31 0004C          BRW     125$
                         00000000G   EF   D4 0004F 1$:      CLRL    INPUT_VALUE                    ; 4926
                                  0000V   31 00055          BRW     126$
                         00000000G   EF   D4 00058 2$:      CLRL    INPUT_VALUE                    ; 4930
                                  0000V   31 0005E          BRW     126$
                    03 00000000G   EF      00   E0 00061 3$:      BBS     #0,FULL_CHOICE,.+3             ; 4936
                                  0000V   31 00069          BRW     9$
                         00000000      8F   DF 0006C          PUSHAL  #0                             ; 4940
                       00000000G   EF      01   FB 00072          CALLS   #1,CLEAR
                    00V00000000G   EF      00   E0 00079          BBS     #0,FULL_PROMPT,6$              ; 4942
                    03 00000000G   EF      00   E0 00081          BBS     #0,TEMP_FULL_PROMPT,.+3
                                  0000V   31 00089          BRW     7$
                         00000000G   EF   9F 0008C 6$:      PUSHAB  SHIFT                          ; 4946
                                      04   DD 00092          PUSHL   #4
                         00000000G   EF   9F 00094          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 0009A          CALLS   #3,PAS$WRITE_STRING
                         FFFF8365    EF   9F 000A1          PUSHAB  C.AOE
                                      02   DD 000A7          PUSHL   #2
                         00000000G   EF   9F 000A9          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 000AF          CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 000B6          PUSHAB  ANSI_REVERSE
                                      04   DD 000BC          PUSHL   #4
                         00000000G   EF   9F 000BE          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 000C4          CALLS   #3,PAS$WRITE_STRING
                         FFFF833F    EF   9F 000CB          PUSHAB  C.AOF
                                      0D   DD 000D1          PUSHL   #13
                         00000000G   EF   9F 000D3          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 000D9          CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 000E0          PUSHAB  SEC_ATTR
                                      16   DD 000E6          PUSHL   #22
                         00000000G   EF   9F 000E8          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 000EE          CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 000F5          PUSHAB  ANSI_RESET
                                      04   DD 000FB          PUSHL   #4
                         00000000G   EF   9F 000FD          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 00103          CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 0010A          PUSHAB  CRLF
                                      02   DD 00110          PUSHL   #2
                         00000000G   EF   9F 00112          PUSHAB  PAS$FV_OUTPUT
                    00000000G   EF        03   FB 00118          CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 0011F          PUSHAB  CRLF_SHIFT
```

```
                                 06   DD 00125          PUSHL   #6
                      00000000G  EF   9F 00127          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0012D          CALLS   #3,PAS$WRITE_STRING
                      FFFF82E6   EF   9F 00134          PUSHAB  C.AOG
                                 10   DD 0013A          PUSHL   #16
                      00000000G  EF   9F 0013C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00142          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 00149          PUSHAB  CRLF_SHIFT
                                 06   DD 0014F          PUSHL   #6
                      00000000G  EF   9F 00151          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00157          CALLS   #3,PAS$WRITE_STRING
                      FFFF82CC   EF   9F 0015E          PUSHAB  C.AOH
                                 0F   DD 00164          PUSHL   #15
                      00000000G  EF   9F 00166          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0016C          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 00173          PUSHAB  CRLF_SHIFT
                                 06   DD 00179          PUSHL   #6
                      00000000G  EF   9F 0017B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00181          CALLS   #3,PAS$WRITE_STRING
                      FFFF82B2   EF   9F 00188          PUSHAB  C.AOI
                                 0C   DD 0018E          PUSHL   #12
                      00000000G  EF   9F 00190          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00196          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 0019D          PUSHAB  CRLF_SHIFT
                                 06   DD 001A3          PUSHL   #6
                      00000000G  EF   9F 001A5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001AB          CALLS   #3,PAS$WRITE_STRING
                      FFFF8294   EF   9F 001B2          PUSHAB  C.AOJ
                                 0C   DD 001B8          PUSHL   #12
                      00000000G  EF   9F 001BA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001C0          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 001C7          PUSHAB  CRLF_SHIFT
                                 06   DD 001CD          PUSHL   #6
                      00000000G  EF   9F 001CF          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001D5          CALLS   #3,PAS$WRITE_STRING
                      FFFF8276   EF   9F 001DC          PUSHAB  C.AOK
                                 11   DD 001E2          PUSHL   #17
                      00000000G  EF   9F 001E4          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001EA          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 001F1          PUSHAB  CRLF_SHIFT
                                 06   DD 001F7          PUSHL   #6
                      00000000G  EF   9F 001F9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001FF          CALLS   #3,PAS$WRITE_STRING
                      FFFF8260   EF   9F 00206          PUSHAB  C.AOL
                                 10   DD 0020C          PUSHL   #16
                      00000000G  EF   9F 0020E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00214          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 0021B          PUSHAB  CRLF_SHIFT
                                 06   DD 00221          PUSHL   #6
                      00000000G  EF   9F 00223          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00229          CALLS   #3,PAS$WRITE_STRING
                      FFFF8246   EF   9F 00230          PUSHAB  C.AOM
                                 0F   DD 00236          PUSHL   #15
                      00000000G  EF   9F 00238          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0023E          CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 00245          PUSHAB  CRLF
                                 02   DD 0024B          PUSHL   #2
```

```
                    00000000G  EF   9F 0024D          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 00253          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0025A          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      01 FB 00260          CALLS   #1,PAS$WRITELN2
                             0000V 31 00267          BRW     14$
                    00000000G  EF   9F 0026A 7$:      PUSHAB  SHIFT                                   ; 4969
                               04   DD 00270          PUSHL   #4
                    00000000G  EF   9F 00272          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 00278          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0027F          PUSHAB  QUES_HINT
                               1F   DD 00285          PUSHL   #31
                    00000000G  EF   9F 00287          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 0028D          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00294          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      01 FB 0029A          CALLS   #1,PAS$WRITELN2
                             0000V 31 002A1          BRW     14$
                    00000000   8F   DF 002A4 9$:      PUSHAL  #0                                      ; 4977
                00000000G  EF      01 FB 002AA          CALLS   #1,CLEAR
             00V00000000G  EF      00 E0 002B1          BBS     #0,FULL_PROMPT,11$                      ; 4979
             03 00000000G  EF      00 E0 002B9          BBS     #0,TEMP_FULL_PROMPT,.+3
                             0000V 31 002C1          BRW     12$
                    00000000G  EF   9F 002C4 11$:     PUSHAB  SHIFT                                   ; 4983
                               04   DD 002CA          PUSHL   #4
                    00000000G  EF   9F 002CC          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 002D2          CALLS   #3,PAS$WRITE_STRING
                    FFFF81AD   EF   9F 002D9          PUSHAB  C.AON
                               02   DD 002DF          PUSHL   #2
                    00000000G  EF   9F 002E1          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 002E7          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 002EE          PUSHAB  ANSI_REVERSE
                               04   DD 002F4          PUSHL   #4
                    00000000G  EF   9F 002F6          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 002FC          CALLS   #3,PAS$WRITE_STRING
                    FFFF8187   EF   9F 00303          PUSHAB  C.AOO
                               0F   DD 00309          PUSHL   #15
                    00000000G  EF   9F 0030B          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 00311          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00318          PUSHAB  SEC_ATTR
                               16   DD 0031E          PUSHL   #22
                    00000000G  EF   9F 00320          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 00326          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0032D          PUSHAB  ANSI_RESET
                               04   DD 00333          PUSHL   #4
                    00000000G  EF   9F 00335          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 0033B          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00342          PUSHAB  CRLF
                               02   DD 00348          PUSHL   #2
                    00000000G  EF   9F 0034A          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      03 FB 00350          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 00357          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF      01 FB 0035D          CALLS   #1,PAS$WRITELN2
                    000000FC   8F   DD 00364          PUSHL   #252                                    ; 4990
                               07   DD 0036A          PUSHL   #7
                               04   DD 0036C          PUSHL   #4
                    00000000G  EF   9F 0036E          PUSHAB  SYS$OUTPUT_NAME
                               0B   DD 00374          PUSHL   #11
                               01   DD 00376          PUSHL   #1
```

```
                    00000000G  EF  9F 00378          PUSHAB  FDL_DEST
00000000G  EF                   07  FB 0037E          CALLS   #7,PAS$OPEN2
                    00000000G  EF  9F 00385          PUSHAB  FDL_DEST                                ; 4992
00000000G  EF                   01  FB 0038B          CALLS   #1,PAS$REWRITE2
                    00000000G  EF  9F 00392          PUSHAB  TEST                                    ; 4994
00000000G  EF                   01  FB 00398          CALLS   #1,SHOW_PRIMARY_SECTION
                    00000000G  EF  9F 0039F          PUSHAB  FDL_DEST                                ; 4996
00000000G  EF                   01  FB 003A5          CALLS   #1,PAS$CLOSE2
                                00V 11  003AC          BRB     14$
                    00000000G  EF  9F 003AE  12$:      PUSHAB  SHIFT                                   ; 5002
                                04  DD 003B4          PUSHL   #4
                    00000000G  EF  9F 003B6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 003BC          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 003C3          PUSHAB  QUES_HINT
                                1F  DD 003C9          PUSHL   #31
                    00000000G  EF  9F 003CB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 003D1          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 003D8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   01  FB 003DE          CALLS   #1,PAS$WRITELN2
                    00000000G  EF  9F 003E5  14$:      PUSHAB  SHIFT                                   ; 5006
                                04  DD 003EB          PUSHL   #4
                    00000000G  EF  9F 003ED          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 003F3          CALLS   #3,PAS$WRITE_STRING
                    FFFF80A0   EF  9F 003FA          PUSHAB  C.AOP
                                21  DD 00400          PUSHL   #33
                    00000000G  EF  9F 00402          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 00408          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 0040F          PUSHAB  ANSI_REVERSE
                                04  DD 00415          PUSHL   #4
                    00000000G  EF  9F 00417          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 0041D          CALLS   #3,PAS$WRITE_STRING
                    FFFF809A   EF  9F 00424          PUSHAB  C.AOQ
                                03  DD 0042A          PUSHL   #3
                    00000000G  EF  9F 0042C          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 00432          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 00439          PUSHAB  ANSI_RESET
                                04  DD 0043F          PUSHL   #4
                    00000000G  EF  9F 00441          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 00447          CALLS   #3,PAS$WRITE_STRING
                    FFFF8074   EF  9F 0044E          PUSHAB  C.AOR
                                03  DD 00454          PUSHL   #3
                    00000000G  EF  9F 00456          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                   03  FB 0045C          CALLS   #3,PAS$WRITE_STRING
                    000000^0   8F  DF 00463          PUSHAL  #0                                      ; 5008
                          00   8F  9F 00469          PUSHAB  #0
      F4    AD 00000000G  EF  9E 0046C          MOVAB   EDF$AB_ACCESS_TABLE_STA,-12(FP)
                        F4    AD  9F 00474          PUSHAB  -12(FP)
      F0    AD 00000000G  EF  9E 00477          MOVAB   EDF$AB_ACCESS_TABLE_KEY,-16(FP)
                        F0    AD  9F 0047F          PUSHAB  -16(FP)
00000000G  EF                   04  FB 00482          CALLS   #4,PARSE_INPUT
                              0000V 31  00489          BRW     126$
                    00000000G  EF  D4 0048C  15$:      CLRL    INPUT_VALUE                             ; 5099
                              0000V 31  00492          BRW     126$
                    00000000G  EF  D4 00495  16$:      CLRL    INPUT_VALUE                             ; 5103
                              0000V 31  0049B          BRW     126$
03 00000000G  EF                00  E0 0049E  17$:      BBS     #0,FULL_CHOICE,.+3                      ; 5109
                              0000V 31  004A6          BRW     23$
```

```
                 00000000   8F   DF 004A9          PUSHAL  #0                                      ; 5113
    00000000G    EF         01   FB 004AF          CALLS   #1,CLEAR
00V000000000G    EF         00   E0 004B6          BBS     #0,FULL_PROMPT,20$                      ; 5115
03 00000000G     EF         00   E0 004BE          BBS     #0,TEMP_FULL_PROMPT,.+3
                      0000V 31   004C6            BRW     21$
                 00000000G  EF   9F 004C9 20$:     PUSHAB  SHIFT                                   ; 5119
                            04   DD 004CF          PUSHL   #4
                 00000000G  EF   9F 004D1          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 004D7          CALLS   #3,PAS$WRITE_STRING
                 FFFF7FE8   EF   9F 004DE          PUSHAB  C_AOS
                            02   DD 004E4          PUSHL   #2
                 00000000G  EF   9F 004E6          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 004EC          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 004F3          PUSHAB  ANSI_REVERSE
                            04   DD 004F9          PUSHL   #4
                 00000000G  EF   9F 004FB          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 00501          CALLS   #3,PAS$WRITE_STRING
                 FFFF7FC2   EF   9F 00508          PUSHAB  C_AOT
                            0C   DD 0050E          PUSHL   #12
                 00000000G  EF   9F 00510          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 00516          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 0051D          PUSHAB  ACTIVE_AREA
    00000000G    EF         01   FB 00523          CALLS   #1,NUM_LEN
                            50   DD 0052A          PUSHL   R0
                 00000000G  EF   DD 0052C          PUSHL   ACTIVE_AREA
                 00000000G  EF   9F 00532          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 00538          CALLS   #3,PAS$WRITE_INTEGER
                 00000000G  EF   9F 0053F          PUSHAB  SEC_ATTR
                            16   DD 00545          PUSHL   #22
                 00000000G  EF   9F 00547          PUSHAB  PAS$FV_OUTPUT
    00000000G    EF         03   FB 0054D          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 00554          PUSHAB  ANSI_RESET
                            04   DD 0055A          PUSHL   #4
                 00000000G  EF   9F 0055C          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 00562          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 00569          PUSHAB  CRLF
                            02   DD 0056F          PUSHL   #2
                 00000000G  EF   9F 00571          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 00577          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 0057E          PUSHAB  CRLF_SHIFT
                            06   DD 00584          PUSHL   #6
                 00000000G  EF   9F 00586          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 0058C          CALLS   #3,PAS$WRITE_STRING
                 FFFF7F43   EF   9F 00593          PUSHAB  C_AOU
                            12   DD 00599          PUSHL   #18
                 00000000G  EF   9F 0059B          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 005A1          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 005A8          PUSHAB  CRLF_SHIFT
                            06   DD 005AE          PUSHL   #6
                 00000000G  EF   9F 005B0          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 005B6          CALLS   #3,PAS$WRITE_STRING
                 FFFF7F2D   EF   9F 005BD          PUSHAB  C_AOV
                            1A   DD 005C3          PUSHL   #26
                 00000000G  EF   9F 005C5          PUSHAB  PAS$FV_OUTPUT
00000000G    EF             03   FB 005CB          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 005D2          PUSHAB  CRLF_SHIFT
                            06   DD 005D8          PUSHL   #6
```

```
                                 Generated Code

                    00000000G   EF  9F 005DA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 005E0          CALLS   #3,PAS$WRITE_STRING
                    FFFF7F1F    EF  9F 005E7          PUSHAB  C.AOW
                                13  DD 005ED          PUSHL   #19
                    00000000G   EF  9F 005EF          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 005F5          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 005FC          PUSHAB  CRLF_SHIFT
                                06  DD 00602          PUSHL   #6
                    00000000G   EF  9F 00604          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 0060A          CALLS   #3,PAS$WRITE_STRING
                    FFFF7F09    EF  9F 00611          PUSHAB  C.AOX
                                12  DD 00617          PUSHL   #18
                    00000000G   EF  9F 00619          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 0061F          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 00626          PUSHAB  CRLF_SHIFT
                                06  DD 0062C          PUSHL   #6
                    00000000G   EF  9F 0062E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 00634          CALLS   #3,PAS$WRITE_STRING
                    FFFF7EF3    EF  9F 0063B          PUSHAB  C.AOY
                                18  DD 00641          PUSHL   #24
                    00000000G   EF  9F 00643          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 00649          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 00650          PUSHAB  CRLF_SHIFT
                                06  DD 00656          PUSHL   #6
                    00000000G   EF  9F 00658          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 0065E          CALLS   #3,PAS$WRITE_STRING
                    FFFF7EE1    EF  9F 00665          PUSHAB  C.AOZ
                                11  DD 0066B          PUSHL   #17
                    00000000G   EF  9F 0066D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 00673          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 0067A          PUSHAB  CRLF_SHIFT
                                06  DD 00680          PUSHL   #6
                    00000000G   EF  9F 00682          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 00688          CALLS   #3,PAS$WRITE_STRING
                    FFFF7ECB    EF  9F 0068F          PUSHAB  C.APA
                                1C  DD 00695          PUSHL   #28
                    00000000G   EF  9F 00697          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 0069D          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 006A4          PUSHAB  CRLF_SHIFT
                                06  DD 006AA          PUSHL   #6
                    00000000G   EF  9F 006AC          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 006B2          CALLS   #3,PAS$WRITE_STRING
                    FFFF7EBD    EF  9F 006B9          PUSHAB  C.APB
                                0F  DD 006BF          PUSHL   #15
                    00000000G   EF  9F 006C1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 006C7          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 006CE          PUSHAB  CRLF
                                02  DD 006D4          PUSHL   #2
                    00000000G   EF  9F 006D6          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 006DC          CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF  9F 006E3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01  FB 006E9          CALLS   #1,PAS$WRITELN2
                                00V 11 006F0          BRB     22$
                    00000000G   EF  9F 006F2  21$:     PUSHAB  SHIFT                   ; 5145
                                04  DD 006F8          PUSHL   #4
                    00000000G   EF  9F 006FA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03  FB 00700          CALLS   #3,PAS$WRITE_STRING
```

```
                    00000000G  EF    9F 00707          PUSHAB  QUES_HINT
                               1F    DD 0070D          PUSHL   #31
                    00000000G  EF    9F 0070F          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 00715          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 0071C          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              01 00722          CALLS   #1,PAS$WRITELN2
                             0000V   31 00729 22$:     BRW     28$
                    00000000   8F    DF 0072C 23$:     PUSHAL  #0                                      ; 5153
          00000000G  EF              01 00732          CALLS   #1,CLEAR
       00V00000000G  EF              00 00739          BBS     #0,FULL_PROMPT,25$                      ; 5155
       03 00000000G  EF              00 00741          BBS     #0,TEMP_FULL_PROMPT,.+3
                             0000V   31 00749          BRW     26$
                    00000000G  EF    9F 0074C 25$:     PUSHAB  SHIFT                                   ; 5159
                               04    DD 00752          PUSHL   #4
                    00000000G  EF    9F 00754          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 0075A          CALLS   #3,PAS$WRITE_STRING
                    FFFF7E25   EF    9F 00761          PUSHAB  C.APC
                               02    DD 00767          PUSHL   #2
                    00000000G  EF    9F 00769          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 0076F          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 00776          PUSHAB  ANSI_REVERSE
                               04    DD 0077C          PUSHL   #4
                    00000000G  EF    9F 0077E          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 00784          CALLS   #3,PAS$WRITE_STRING
                    FFFF7DFF   EF    9F 0078B          PUSHAB  C.APD
                               0E    DD 00791          PUSHL   #14
                    00000000G  EF    9F 00793          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 00799          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 007A0          PUSHAB  ACTIVE_AREA
          00000000G  EF              01 007A6          CALLS   #1,NUM_LEN
                               50    DD 007AD          PUSHL   R0
                    00000000G  EF    DD 007AF          PUSHL   ACTIVE_AREA
                    00000000G  EF    9F 007B5          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 007BB          CALLS   #3,PAS$WRITE_INTEGER
                    00000000G  EF    9F 007C2          PUSHAB  SEC_ATTR
                               16    DD 007C8          PUSHL   #22
                    00000000G  EF    9F 007CA          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 007D0          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 007D7          PUSHAB  ANSI_RESET
                               04    DD 007DD          PUSHL   #4
                    00000000G  EF    9F 007DF          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 007E5          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 007EC          PUSHAB  CRLF
                               02    DD 007F2          PUSHL   #2
                    00000000G  EF    9F 007F4          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03 007FA          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 00801          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              01 00807          CALLS   #1,PAS$WRITELN2
                    000000FC   8F    DD 0080E          PUSHL   #252                                    ; 5168
                               07    DD 00814          PUSHL   #7
                               04    DD 00816          PUSHL   #4
                    00000000G  EF    9F 00818          PUSHAB  SYS$OUTPUT_NAME
                               0B    DD 0081E          PUSHL   #11
                               01    DD 00820          PUSHL   #1
                    00000000G  EF    9F 00822          PUSHAB  FDL_DEST
          00000000G  EF              07 00828          CALLS   #7,PAS$OPEN2
                    00000000G  EF    9F 0082F          PUSHAB  FDL_DEST                                ; 5170
```

Generated Code

```
                                    01   FB 00835          CALLS   #1,PAS$REWRITE2
                          00000000G EF   9F 0083C          PUSHAB  TEST                                    ; 5172
00000000G EF                        01   FB 00842          CALLS   #1,SHOW_PRIMARY_SECTION
                          00000000G EF   9F 00849          PUSHAB  FDL_DEST                                ; 5174
00000000G EF                        01   FB 0084F          CALLS   #1,PAS$CLOSE2
                                    00V  11 00856          BRB     28$
                          00000000G EF   9F 00858 26$:     PUSHAB  SHIFT                                   ; 5180
                                    04   DD 0085E          PUSHL   #4
                          00000000G EF   9F 00860          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 00866          CALLS   #3,PAS$WRITE_STRING
                          00000000G EF   9F 0086D          PUSHAB  QUES_HINT
                                    1F   DD 00873          PUSHL   #31
                          00000000G EF   9F 00875          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 0087B          CALLS   #3,PAS$WRITE_STRING
                          00000000G EF   9F 00882          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        01   FB 00888          CALLS   #1,PAS$WRITELN2
                          00000000G EF   9F 0088F 28$:     PUSHAB  SHIFT                                   ; 5184
                                    04   DD 00895          PUSHL   #4
                          00000000G EF   9F 00897          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 0089D          CALLS   #3,PAS$WRITE_STRING
                          FFFF7CF6  EF   9F 008A4          PUSHAB  C.APE
                                    0B   DD 008AA          PUSHL   #11
                          00000000G EF   9F 008AC          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 008B2          CALLS   #3,PAS$WRITE_STRING
                          00000000G EF   9F 008B9          PUSHAB  ACTIVE_AREA
00000000G EF                        01   FB 008BF          CALLS   #1,NUM_LEN
                                    50   DD 008C6          PUSHL   R0
                          00000000G EF   DD 008C8          PUSHL   ACTIVE_AREA
                          00000000G EF   9F 008CE          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 008D4          CALLS   #3,PAS$WRITE_INTEGER
                          FFFF7CCB  EF   9F 008DB          PUSHAB  C.APF
                                    15   DD 008E1          PUSHL   #21
                          00000000G EF   9F 008E3          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 008E9          CALLS   #3,PAS$WRITE_STRING
                          00000000G EF   9F 008F0          PUSHAB  ANSI_REVERSE
                                    04   DD 008F6          PUSHL   #4
                          00000000G EF   9F 008F8          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 008FE          CALLS   #3,PAS$WRITE_STRING
                          FFFF7CB9  EF   9F 00905          PUSHAB  C.APG
                                    03   DD 0090B          PUSHL   #3
                          00000000G EF   9F 0090D          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 00913          CALLS   #3,PAS$WRITE_STRING
                          00000000G EF   9F 0091A          PUSHAB  ANSI_RESET
                                    04   DD 00920          PUSHL   #4
                          00000000G EF   9F 00922          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 00928          CALLS   #3,PAS$WRITE_STRING
                          FFFF7C93  EF   9F 0092F          PUSHAB  C.APH
                                    03   DD 00935          PUSHL   #3
                          00000000G EF   9F 00937          PUSHAB  PAS$FV_OUTPUT
00000000G EF                        03   FB 0093D          CALLS   #3,PAS$WRITE_STRING
                          00000000  8F   DF 00944          PUSHAL  #0                                      ; 5187
                                 00  8F   9F 0094A          PUSHAB  #0
      F4   AD 00000000G EF   9E 0094D          MOVAB   EDF$AB_AREA_TABLE_STA,-12(FP)
                                 F4   AD   9F 00955          PUSHAB  -12(FP)
      F0   AD 00000000G EF   9E 00958          MOVAB   EDF$AB_AREA_TABLE_KEY,-16(FP)
                                 F0   AD   9F 00960          PUSHAB  -16(FP)
00000000G EF                        04   FB 00963          CALLS   #4,PARSE_INPUT
```

```
                    0000V  31 0096A          BRW     126$
03 00000000G  EF      00   E0 0096D 29$:     BBS     #0,FULL_CHOICE,..+3                    ; 5200
                    0000V  31 00975          BRW     35$
              00000000  8F   DF 00978          PUSHAL  #0                                     ; 5204
   00000000G  EF      01   FB 0097E          CALLS   #1,CLEAR
00V00000000G  EF      00   E0 00985          BBS     #0,FULL_PROMPT,32$                     ; 5206
03 00000000G  EF      00   E0 0098D          BBS     #0,TEMP_FULL_PROMPT,..+3
                    0000V  31 00995          BRW     33$
                00000000G  EF  9F 00998 32$:  PUSHAB  SHIFT                                  ; 5210
                      04   DD 0099E          PUSHL   #4
                00000000G  EF  9F 009A0      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 009A6          CALLS   #3,PAS$WRITE_STRING
                FFFF7C19   EF  9F 009AD      PUSHAB  C.API
                      02   DD 009B3          PUSHL   #2
                00000000G  EF  9F 009B5      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 009BB          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 009C2      PUSHAB  ANSI_REVERSE
                      04   DD 009C8          PUSHL   #4
                00000000G  EF  9F 009CA      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 009D0          CALLS   #3,PAS$WRITE_STRING
                FFFF7BF3   EF  9F 009D7      PUSHAB  C.APJ
                      0E   DD 009DD          PUSHL   #14
                00000000G  EF  9F 009DF      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 009E5          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 009EC      PUSHAB  SEC_ATTR
                      16   DD 009F2          PUSHL   #22
                00000000G  EF  9F 009F4      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 009FA          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 00A01      PUSHAB  ANSI_RESET
                      04   DD 00A07          PUSHL   #4
                00000000G  EF  9F 00A09      PUSHAB  PAS$FV_OUTPUT
   00000000G  EF      03   FB 00A0F          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 00A16      PUSHAB  CRLF
                      02   DD 00A1C          PUSHL   #2
                00000000G  EF  9F 00A1E      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A24          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 00A2B      PUSHAB  CRLF_SHIFT
                      06   DD 00A31          PUSHL   #6
                00000000G  EF  9F 00A33      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A39          CALLS   #3,PAS$WRITE_STRING
                FFFF7B9A   EF  9F 00A40      PUSHAB  C.APK
                      24   DD 00A46          PUSHL   #36
                00000000G  EF  9F 00A48      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A4E          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 00A55      PUSHAB  CRLF_SHIFT
                      06   DD 00A5B          PUSHL   #6
                00000000G  EF  9F 00A5D      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A63          CALLS   #3,PAS$WRITE_STRING
                FFFF7B94   EF  9F 00A6A      PUSHAB  C.APL
                      2A   DD 00A70          PUSHL   #42
                00000000G  EF  9F 00A72      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A78          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF  9F 00A7F      PUSHAB  CRLF_SHIFT
                      06   DD 00A85          PUSHL   #6
                00000000G  EF  9F 00A87      PUSHAB  PAS$FV_OUTPUT
00000000G  EF         03   FB 00A8D          CALLS   #3,PAS$WRITE_STRING
                FFFF7B96   EF  9F 00A94      PUSHAB  C.APM
```

```
                                 26   DD 00A9A          PUSHL   #38
                    00000000G    EF   9F 00A9C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00AA2          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00AA9          PUSHAB  CRLF_SHIFT
                                 06   DD 00AAF          PUSHL   #6
                    00000000G    EF   9F 00AB1          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00AB7          CALLS   #3,PAS$WRITE_STRING
                    FFFF7B94     EF   9F 00ABE          PUSHAB  C.APN
                                 28   DD 00AC4          PUSHL   #40
                    00000000G    EF   9F 00AC6          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00ACC          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00AD3          PUSHAB  CRLF_SHIFT
                                 06   DD 00AD9          PUSHL   #6
                    00000000G    EF   9F 00ADB          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00AE1          CALLS   #3,PAS$WRITE_STRING
                    FFFF7B92     EF   9F 00AE8          PUSHAB  C.APO
                                 2A   DD 00AEE          PUSHL   #42
                    00000000G    EF   9F 00AF0          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00AF6          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00AFD          PUSHAB  CRLF_SHIFT
                                 06   DD 00B03          PUSHL   #6
                    00000000G    EF   9F 00B05          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B0B          CALLS   #3,PAS$WRITE_STRING
                    FFFF7B94     EF   9F 00B12          PUSHAB  C.APP
                                 2A   DD 00B18          PUSHL   #42
                    00000000G    EF   9F 00B1A          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B20          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00B27          PUSHAB  CRLF_SHIFT
                                 06   DD 00B2D          PUSHL   #6
                    00000000G    EF   9F 00B2F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B35          CALLS   #3,PAS$WRITE_STRING
                    FFFF7B96     EF   9F 00B3C          PUSHAB  C.APQ
                                 2C   DD 00B42          PUSHL   #44
                    00000000G    EF   9F 00B44          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B4A          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00B51          PUSHAB  CRLF_SHIFT
                                 06   DD 00B57          PUSHL   #6
                    00000000G    EF   9F 00B59          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B5F          CALLS   #3,PAS$WRITE_STRING
                    FFFF7B98     EF   9F 00B66          PUSHAB  C.APR
                                 33   DD 00B6C          PUSHL   #51
                    00000000G    EF   9F 00B6E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B74          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00B7B          PUSHAB  CRLF_SHIFT
                                 06   DD 00B81          PUSHL   #6
                    00000000G    EF   9F 00B83          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B89          CALLS   #3,PAS$WRITE_STRING
                    FFFF7BA2     EF   9F 00B90          PUSHAB  C.APS
                                 29   DD 00B96          PUSHL   #41
                    00000000G    EF   9F 00B98          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00B9E          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 00BA5          PUSHAB  CRLF_SHIFT
                                 06   DD 00BAB          PUSHL   #6
                    00000000G    EF   9F 00BAD          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 00BB3          CALLS   #3,PAS$WRITE_STRING
                    FFFF7BA4     EF   9F 00BBA          PUSHAB  C.APT
                                 2C   DD 00BC0          PUSHL   #44
```

```
                          Generated Code

                00000000G EF 9F 00BC2          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00BC8          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00BCF          PUSHAB  CRLF_SHIFT
                             06 DD 00BD5          PUSHL   #6
                00000000G EF 9F 00BD7          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00BDD          CALLS   #3,PAS$WRITE_STRING
                FFFF7BA6  EF 9F 00BE4          PUSHAB  C.APU
                             2E DD 00BEA          PUSHL   #46
                00000000G EF 9F 00BEC          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00BF2          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00BF9          PUSHAB  CRLF_SHIFT
                             06 DD 00BFF          PUSHL   #6
                00000000G EF 9F 00C01          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C07          CALLS   #3,PAS$WRITE_STRING
                FFFF7BAC  EF 9F 00C0E          PUSHAB  C.APV
                             2B DD 00C14          PUSHL   #43
                00000000G EF 9F 00C16          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C1C          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00C23          PUSHAB  CRLF_SHIFT
                             06 DD 00C29          PUSHL   #6
                00000000G EF 9F 00C2B          PUSHAB  PAS$FV_OUTPUT
00J000000G EF               03 FB 00C31          CALLS   #3,PAS$WRITE_STRING
                FFFF7BAE  EF 9F 00C38          PUSHAB  C.APW
                             2C DD 00C3E          PUSHL   #44
                00000000G EF 9F 00C40          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C46          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00C4D          PUSHAB  CRLF_SHIFT
                             06 DD 00C53          PUSHL   #6
                00000000G EF 9F 00C55          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C5B          CALLS   #3,PAS$WRITE_STRING
                FFFF7BB0  EF 9F 00C62          PUSHAB  C.APX
                             27 DD 00C68          PUSHL   #39
                00000000G EF 9F 00C6A          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C70          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00C77          PUSHAB  CRLF_SHIFT
                             06 DD 00C7D          PUSHL   #6
                00000000G EF 9F 00C7F          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C85          CALLS   #3,PAS$WRITE_STRING
                FFFF7BAE  EF 9F 00C8C          PUSHAB  C.APY
                             2F DD 00C92          PUSHL   #47
                00000000G EF 9F 00C94          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00C9A          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00CA1          PUSHAB  CRLF_SHIFT
                             06 DD 00CA7          PUSHL   #6
                00000000G EF 9F 00CA9          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00CAF          CALLS   #3,PAS$WRITE_STRING
                FFFF7BB4  EF 9F 00CB6          PUSHAB  C.APZ
                             2C DD 00CBC          PUSHL   #44
                00000000G EF 9F 00CBE          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00CC4          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 00CCB          PUSHAB  CRLF_SHIFT
                             06 DD 00CD1          PUSHL   #6
                00000000G EF 9F 00CD3          PUSHAB  PAS$FV_OUTPUT
00000000G EF                03 FB 00CD9          CALLS   #3,PAS$WRITE_STRING
                FFFF7BB6  EF 9F 00CE0          PUSHAB  C.AQA
                             18 DD 00CE6          PUSHL   #24
                00000000G EF 9F 00CE8          PUSHAB  PAS$FV_OUTPUT
```

```
                    00000000G  EF                03  FB 00CEE           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00CF5           PUSHAB  CRLF
                                                     02  DD 00CFB           PUSHL   #2
                                          00000000G  EF  9F 00CFD           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00D03           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00D0A           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                01  FB 00D10           CALLS   #1,PAS$WRITELN2
                                               0000V 31  00D17           BRW     40$
                                          00000000G  EF  9F 00D1A 33$:      PUSHAB  SHIFT                                   ; 5254
                                                     04  DD 00D20           PUSHL   #4
                                          00000000G  EF  9F 00D22           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00D28           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00D2F           PUSHAB  QUES_HINT
                                                     1F  DD 00D35           PUSHL   #31
                                          00000000G  EF  9F 00D37           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00D3D           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00D44           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                01  FB 00D4A           CALLS   #1,PAS$WRITELN2
                                               0000V 31  00D51           BRW     40$
                                          00000000   8F  DF 00D54 35$:      PUSHAL  #0                                      ; 5262
                 00000000G  EF                   01  FB 00D5A           CALLS   #1,CLEAR
              00V00000000G  EF                   00  E0 00D61           BBS     #0,FULL_PROMPT,37$                      ; 5264
              03 00000000G  cF                   00  E0 00D69           BBS     #0,TEMP_FULL_PROMPT,.+3
                                               0000V 31  00D71           BRW     38$
                                          00000000G  EF  9F 00D74 37$:      PUSHAB  SHIFT                                   ; 5268
                                                     04  DD 00D7A           PUSHL   #4
                                          00000000G  EF  9F 00D7C           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00D82           CALLS   #3,PAS$WRITE_STRING
                                          FFFF7B25   EF  9F 00D89           PUSHAB  C.AQB
                                                     02  DD 00D8F           PUSHL   #2
                                          00000000G  EF  9F 00D91           PUSHAB  PAS$FV_OUTPUT
                 00C00000G  EF                   03  FB 00D97           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00D9E           PUSHAB  ANSI_REVERSE
                                                     04  DD 00DA4           PUSHL   #4
                                          00000000G  EF  9F 00DA6           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00DAC           CALLS   #3,PAS$WRITE_STRING
                                          FFFF7AFF   EF  9F 00DB3           PUSHAB  C.AQC
                                                     10  DD 00DB9           PUSHL   #16
                                          00000000G  EF  9F 00DBB           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00DC1           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00DC8           PUSHAB  SEC_ATTR
                                                     16  DD 00DCE           PUSHL   #22
                                          00000000G  EF  9F 00DD0           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00DD6           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00DDD           PUSHAB  ANSI_RESET
                                                     04  DD 00DE3           PUSHL   #4
                                          00000000G  EF  9F 00DE5           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00DEB           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00DF2           PUSHAB  CRLF
                                                     02  DD 00DF8           PUSHL   #2
                                          00000000G  EF  9F 00DFA           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                03  FB 00E00           CALLS   #3,PAS$WRITE_STRING
                                          00000000G  EF  9F 00E07           PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                01  FB 00E0D           CALLS   #1,PAS$WRITELN2
                                          000000FC   8F  DD 00E14           PUSHL   #252                                    ; 5275
                                                     07  DD 00E1A           PUSHL   #7
                                                     04  DD 00E1C           PUSHL   #4
```

```
                 00000000G   EF   9F  00E1E            PUSHAB  SYS$OUTPUT_NAME
                             0B   DD  00E24            PUSHL   #11
                             01   DD  00E26            PUSHL   #1
                 00000000G   EF   9F  00E28            PUSHAB  FDL_DEST
00000000G   EF               07   FB  00E2E            CALLS   #7,PAS$OPEN2
                 00000000G   EF   9F  00E35            PUSHAB  FDL_DEST                                ; 5277
00000000G   EF               01   FB  00E3B            CALLS   #1,PAS$REWRITE2
                 00000000G   EF   9F  00E42            PUSHAB  TEST                                    ; 5279
00000000G   EF               01   FB  00E48            CALLS   #1,SHOW_PRIMARY_SECTION
                 00000000G   EF   9F  00E4F            PUSHAB  FDL_DEST                                ; 5281
00000000G   EF               01   FB  00E55            CALLS   #1,PAS$CLOSE2
                             00V  11  00E5C            BRB     40$
                 00000000G   EF   9F  00E5E  38$:      PUSHAB  SHIFT                                   ; 5287
                             04   DD  00E64            PUSHL   #4
                 00000000G   EF   9F  00E66            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00E6C            CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F  00E73            PUSHAB  QUES_HINT
                             1F   DD  00E79            PUSHL   #31
                 00000000G   EF   9F  00E7B            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00E81            CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F  00E88            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               01   FB  00E8E            CALLS   #1,PAS$WRITELN2
                 00000000G   EF   9F  00E95  40$:      PUSHAB  SHIFT                                   ; 5291
                             04   DD  00E9B            PUSHL   #4
                 00000000G   EF   9F  00E9D            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00EA3            CALLS   #3,PAS$WRITE_STRING
                 FFFF7A18    EF   9F  00EAA            PUSHAB  C_AQD
                             22   DD  00EB0            PUSHL   #34
                 00000000G   EF   9F  00EB2            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00EB8            CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F  00EBF            PUSHAB  ANSI_REVERSE
                             04   DD  00EC5            PUSHL   #4
                 00000000G   EF   9F  00EC7            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00ECD            CALLS   #3,PAS$WRITE_STRING
                 FFFF7A12    EF   9F  00ED4            PUSHAB  C_AQE
                             03   DD  00EDA            PUSHL   #3
                 00000000G   EF   9F  00EDC            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00EE2            CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF   9F  00EE9            PUSHAB  ANSI_RESET
                             04   DD  00EEF            PUSHL   #4
                 00000000G   EF   9F  00EF1            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00EF7            CALLS   #3,PAS$WRITE_STRING
                 FFFF79EC    EF   9F  00EFE            PUSHAB  C_AQF
                             03   DD  00F04            PUSHL   #3
                 00000000G   EF   9F  00F06            PUSHAB  PAS$FV_OUTPUT
00000000G   EF               03   FB  00F0C            CALLS   #3,PAS$WRITE_STRING
                 00000000    8F   DF  00F13            PUSHAL  #0                                      ; 5293
                       00    8F   9F  00F19            PUSHAB  #0
       F4   AD 00000000G     EF   9E  00F1C            MOVAB   EDF$AB_CONNECT_TABLE_STA,-12(FP)
                       F4    AD   9F  00F24            PUSHAB  -12(FP)
       F0   AD 00000000G     EF   9E  00F27            MOVAB   EDF$AB_CONNECT_TABLE_KEY,-16(FP)
                       F0    AD   9F  00F2F            PUSHAB  -16(FP)
00000000G   EF               04   FB  00F32            CALLS   #4,PARSE_INPUT
                         0000V  31  00F39            BRW     126$
03 00000000G   EF            00   E0  00F3C  41$:      BBS     #0,FULL_CHOICE,.+3                      ; 5306
                         0000V  31  00F44            BRW     47$
                 00000000    8F   DF  00F47            PUSHAL  #0                                      ; 5310
```

```
                    00000000G  EF              01  FB 00F4D          CALLS   #1,CLEAR
                00V 00000000G  EF              00  E0 00F54          BBS     #0,FULL_PROMPT,44$                          ; 5312
                03  00000000G  EF              00  E0 00F5C          BBS     #0,TEMP_FULL_PROMPT,..+3
                                             0000V  31 00F64          BRW     45$
                                    00000000G  EF  9F 00F67 44$:      PUSHAB  SHIFT                                       ; 5316
                                               04  DD 00F6D          PUSHL   #4
                                    00000000G  EF  9F 00F6F          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00F75          CALLS   #3,PAS$WRITE_STRING
                                    FFFF7972   EF  9F 00F7C          PUSHAB  C.AQG
                                               02  DD 00F82          PUSHL   #2
                                    00000000G  EF  9F 00F84          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00F8A          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 00F91          PUSHAB  ANSI_REVERSE
                                               04  DD 00F97          PUSHL   #4
                                    00000000G  EF  9F 00F99          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00F9F          CALLS   #3,PAS$WRITE_STRING
                                    FFFF794C   EF  9F 00FA6          PUSHAB  C.AQH
                                               0B  DD 00FAC          PUSHL   #11
                                    00000000G  EF  9F 00FAE          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00FB4          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 00FBB          PUSHAB  SEC_ATTR
                                               16  DD 00FC1          PUSHL   #22
                                    00000000G  EF  9F 00FC3          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00FC9          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 00FD0          PUSHAB  ANSI_RESET
                                               04  DD 00FD6          PUSHL   #4
                                    00000000G  EF  9F 00FD8          PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF              03  FB 00FDE          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 00FE5          PUSHAB  CRLF
                                               02  DD 00FEB          PUSHL   #2
                                    00000000G  EF  9F 00FED          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 00FF3          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 00FFA          PUSHAB  CRLF_SHIFT
                                               06  DD 01000          PUSHL   #6
                                    00000000G  EF  9F 01002          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 01008          CALLS   #3,PAS$WRITE_STRING
                                    FFFF78EF   EF  9F 0100F          PUSHAB  C.AQI
                                               0F  DD 01015          PUSHL   #15
                                    00000000G  EF  9F 01017          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 0101D          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 01024          PUSHAB  CRLF_SHIFT
                                               06  DD 0102A          PUSHL   #6
                                    00000000G  EF  9F 0102C          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 01032          CALLS   #3,PAS$WRITE_STRING
                                    FFFF78D5   EF  9F 01039          PUSHAB  C.AQJ
                                               10  DD 0103F          PUSHL   #16
                                    00000000G  EF  9F 01041          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 01047          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 0104E          PUSHAB  CRLF_SHIFT
                                               06  DD 01054          PUSHL   #6
                                    00000000G  EF  9F 01056          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 0105C          CALLS   #3,PAS$WRITE_STRING
                                    FFFF78BB   EF  9F 01063          PUSHAB  C.AQK
                                               12  DD 01069          PUSHL   #18
                                    00000000G  EF  9F 0106B          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                  03  FB 01071          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 01078          PUSHAB  CRLF_SHIFT
```

```
                                 06   DD 0107E          PUSHL   #6
                      00000000G  EF   9F 01080          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03   FB 01086     CALLS   #3,PAS$WRITE_STRING
                                 FFFF78A5  EF 9F 0108D  PUSHAB  C.AQL
                                 10   DD 01093          PUSHL   #16
                      00000000G  EF   9F 01095          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03   FB 0109B     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 010A2          PUSHAB  CRLF
                                 02   DD 010A8          PUSHL   #2
                      00000000G  EF   9F 010AA          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03   FB 010B0     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 010B7          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        01   FB 010BD     CALLS   #1,PAS$WRITELN2
                           0000V     31   010C4        BRW     52$
                      00000000G  EF   9F 010C7 45$:     PUSHAB  SHIFT                          ; 5333
                                 04   DD 010CD          PUSHL   #4
                      00000000G  EF   9F 010CF          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03   FB 010D5     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 010DC          PUSHAB  QUES_HINT
                                 1F   DD 010E2          PUSHL   #31
                      00000000G  EF   9F 010E4          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03   FB 010EA     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 010F1          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        01   FB 010F7     CALLS   #1,PAS$WRITELN2
                           0000V     31   010FE        BRW     52$
                      00000000   8F   DF 01101 47$:     PUSHAL  #0                             ; 5341
                   00000000G  EF        01   FB 01107  CALLS   #1,CLEAR
             00V00000000G  EF           00   E0 0110E  BBS     #0,FULL_PROMPT,49$             ; 5343
          03 00000000G  EF              00   E0 01116  BBS     #0,TEMP_FULL_PROMPT,..+3
                           0000V     31   0111E        BRW     50$
                      00000000G  EF   9F 01121 49$:     PUSHAB  SHIFT                          ; 5347
                                 04   DD 01127          PUSHL   #4
                      00000000G  EF   9F 01129          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 0112F     CALLS   #3,PAS$WRITE_STRING
                                 FFFF780C  EF 9F 01136  PUSHAB  C.AQM
                                 02   DD 0113C          PUSHL   #2
                      00000000G  EF   9F 0113E          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 01144     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 0114B          PUSHAB  ANSI_REVERSE
                                 04   DD 01151          PUSHL   #4
                      00000000G  EF   9F 01153          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 01159     CALLS   #3,PAS$WRITE_STRING
                                 FFFF77E6  EF 9F 01160  PUSHAB  C.AQN
                                 0D   DD 01166          PUSHL   #13
                      00000000G  EF   9F 01168          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 0116E     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 01175          PUSHAB  SEC_ATTR
                                 16   DD 0117B          PUSHL   #22
                      00000000G  EF   9F 0117D          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 01183     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 0118A          PUSHAB  ANSI_RESET
                                 04   DD 01190          PUSHL   #4
                      00000000G  EF   9F 01192          PUSHAB  PAS$FV_OUTPUT
          00000000G  EF              03   FB 01198     CALLS   #3,PAS$WRITE_STRING
                      00000000G  EF   9F 0119F          PUSHAB  CRLF
                                 02   DD 011A5          PUSHL   #2
                      00000000G  EF   9F 011A7          PUSHAB  PAS$FV_OUTPUT
```

```
                    03    FB 011AD          CALLS   #3,PAS$WRITE_STRING
                00000000G EF 9F 011B4         PUSHAB  PAS$FV_OUTPUT
00000000G EF                 01    FB 011BA   CALLS   #1,PAS$WRITELN2
                000000FC    8F DD 011C1       PUSHL   #252                                    ; 5354
                            07 DD 011C7       PUSHL   #7
                            04 DD 011C9       PUSHL   #4
                00000000G   EF 9F 011CB       PUSHAB  SYS$OUTPUT_NAME
                            0B DD 011D1       PUSHL   #11
                            01 DD 011D3       PUSHL   #1
                00000000G   EF 9F 011D5       PUSHAB  FDL_DEST
00000000G EF                 07    FB 011DB   CALLS   #7,PAS$OPEN2
                00000000G   EF 9F 011E2       PUSHAB  FDL_DEST                                ; 5356
00000000G EF                 01    FB 011E8   CALLS   #1,PAS$REWRITE2
                00000000G   EF 9F 011EF       PUSHAB  TEST                                    ; 5358
00000000G EF                 01    FB 011F5   CALLS   #1,SHOW_PRIMARY_SECTION
                00000000G   EF 9F 011FC       PUSHAB  FDL_DEST                                ; 5360
00000000G EF                 01    FB 01202   CALLS   #1,PAS$CLOSE2
                            00V 11 01209      BRB     52$
                00000000G   EF 9F 0120B 50$:  PUSHAB  SHIFT                                   ; 5366
                            04 DD 01211       PUSHL   #4
                00000000G   EF 9F 01213       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 01219   CALLS   #3,PAS$WRITE_STRING
                00000000G   EF 9F 01220       PUSHAB  QUES_HINT
                            1F DD 01226       PUSHL   #31
                00000000G   EF 9F 01228       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 0122E   CALLS   #3,PAS$WRITE_STRING
                00000000G   EF 9F 01235       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 01    FB 0123B   CALLS   #1,PAS$WRITELN2
                00000000G   EF 9F 01242 52$:  PUSHAB  SHIFT                                   ; 5370
                            04 DD 01248       PUSHL   #4
                00000000G   EF 9F 0124A       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 01250   CALLS   #3,PAS$WRITE_STRING
                FFFF76FF    EF 9F 01257       PUSHAB  C_AQO
                            1F DD 0125D       PUSHL   #31
                00000000G   EF 9F 0125F       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 01265   CALLS   #3,PAS$WRITE_STRING
                00000000G   EF 9F 0126C       PUSHAB  ANSI_REVERSE
                            04 DD 01272       PUSHL   #4
                00000000G   EF 9F 01274       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 0127A   CALLS   #3,PAS$WRITE_STRING
                FFFF76F5    EF 9F 01281       PUSHAB  C_AQP
                            03 DD 01287       PUSHL   #3
                00000000G   EF 9F 01289       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 0128F   CALLS   #3,PAS$WRITE_STRING
                00000000G   EF 9F 01296       PUSHAB  ANSI_RESET
                            04 DD 0129C       PUSHL   #4
                00000000G   EF 9F 0129E       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 012A4   CALLS   #3,PAS$WRITE_STRING
                FFFF76CF    EF 9F 012AB       PUSHAB  C_AQQ
                            03 DD 012B1       PUSHL   #3
                00000000G   EF 9F 012B3       PUSHAB  PAS$FV_OUTPUT
00000000G EF                 03    FB 012B9   CALLS   #3,PAS$WRITE_STRING
                00000000    8F DF 012C0       PUSHAL  #0                                      ; 5372
                      00    8F 9F 012C6       PUSHAB  #0
        F4    AD 00000000G  EF 9E 012C9       MOVAB   EDF$AB_DATE_TABLE_STA,-12(FP)
                      F4    AD 9F 012D1       PUSHAB  -12(FP)
        F0    AD 00000000G  EF 9E 012D4       MOVAB   EDF$AB_DATE_TABLE_KEY,-16(FP)
```

```
                                 FO    AD   9F 012DC            PUSHAB  -16(FP)
    00000000G   EF                     04   FB 012DF            CALLS   #4,PARSE_INPUT
                                 0000V 31   012E6              BRW     126$
03  00000000G   EF                     00   E0 012E9 53$:       BBS     #0,FULL_CHOICE,..+3                    ; 5385
                                 0000V 31   012F1              BRW     59$
                          00000000   8F   DF 012F4            PUSHAL  #0                                     ; 5389
    00000000G   EF                     01   FB 012FA            CALLS   #1,CLEAR
00V00000000G    EF                     00   E0 01301            BBS     #0,FULL_PROMPT,56$                     ; 5391
03  00000000G   EF                     00   E0 01309            BBS     #0,TEMP_FULL_PROMPT,..+3
                                 0000V 31   01311              BRW     57$
                     00000000G   EF   9F 01314 56$:           PUSHAB  SHIFT                                  ; 5395
                                 04   DD 0131A                PUSHL   #4
                     00000000G   EF   9F 0131C                PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                     03   FB 01322            CALLS   #3,PAS$WRITE_STRING
                     FFFF7655    EF   9F 01329                PUSHAB  C.AQR
                                 02   DD 0132F                PUSHL   #2
                     00000000G   EF   9F 01331                PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                     03   FB 01337            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 0133E                PUSHAB  ANSI_REVERSE
                                 04   DD 01344                PUSHL   #4
                     00000000G   EF   9F 01346                PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                     03   FB 0134C            CALLS   #3,PAS$WRITE_STRING
                     FFFF762F    EF   9F 01353                PUSHAB  C.AQS
                                 0B   DD 01359                PUSHL   #11
                     00000000G   EF   9F 0135B                PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                     03   FB 01361            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 01368                PUSHAB  SEC_ATTR
                                 16   DD 0136E                PUSHL   #22
                     00000000G   EF   9F 01370                PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                     03   FB 01376            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 0137D                PUSHAB  ANSI_RESET
                                 04   DD 01383                PUSHL   #4
                     00000000G   EF   9F 01385                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 0138B            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 01392                PUSHAB  CRLF
                                 02   DD 01398                PUSHL   #2
                     00000000G   EF   9F 0139A                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 013A0            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 013A7                PUSHAB  CRLF_SHIFT
                                 06   DD 013AD                PUSHL   #6
                     00000000G   EF   9F 013AF                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 013B5            CALLS   #3,PAS$WRITE_STRING
                     FFFF75D2    EF   9F 013BC                PUSHAB  C.AQT
                                 2A   DD 013C2                PUSHL   #42
                     00000000G   EF   9F 013C4                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 013CA            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 013D1                PUSHAB  CRLF_SHIFT
                                 06   DD 013D7                PUSHL   #6
                     00000000G   EF   9F 013D9                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 013DF            CALLS   #3,PAS$WRITE_STRING
                     FFFF75D4    EF   9F 013E6                PUSHAB  C.AQU
                                 28   DD 013EC                PUSHL   #40
                     00000000G   EF   9F 013EE                PUSHAB  PAS$FV_OUTPUT
00000000G   EF                     03   FB 013F4            CALLS   #3,PAS$WRITE_STRING
                     00000000G   EF   9F 013FB                PUSHAB  CRLF_SHIFT
                                 06   DD 01401                PUSHL   #6
                     00000000G   EF   9F 01403                PUSHAB  PAS$FV_OUTPUT
```

Generated Code

```
00000000G  EF                03  FB 01409         CALLS   #3,PAS$WRITE_STRING
                 FFFF75D2    EF  9F 01410         PUSHAB  C.AQV
                             24  DD 01416         PUSHL   #36
                 00000000G   EF  9F 01418         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0141E         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 01425         PUSHAB  CRLF_SHIFT
                             06  DD 0142B         PUSHL   #6
                 00000000G   EF  9F 0142D         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 01433         CALLS   #3,PAS$WRITE_STRING
                 FFFF75CC    EF  9F 0143A         PUSHAB  C.AQW
                             2F  DD 01440         PUSHL   #47
                 00000000G   EF  9F 01442         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 01448         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 0144F         PUSHAB  CRLF_SHIFT
                             06  DD 01455         PUSHL   #6
                 00000000G   EF  9F 01457         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0145D         CALLS   #3,PAS$WRITE_STRING
                 FFFF75D2    EF  9F 01464         PUSHAB  C.AQX
                             26  DD 0146A         PUSHL   #38
                 00000000G   EF  9F 0146C         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 01472         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 01479         PUSHAB  CRLF_SHIFT
                             06  DD 0147F         PUSHL   #6
                 00000000G   EF  9F 01481         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 01487         CALLS   #3,PAS$WRITE_STRING
                 FFFF75D0    EF  9F 0148E         PUSHAB  C.AQY
                             2B  DD 01494         PUSHL   #43
                 00000000G   EF  9F 01496         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0149C         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 014A3         PUSHAB  CRLF_SHIFT
                             06  DD 014A9         PUSHL   #6
                 00000000G   EF  9F 014AB         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 014B1         CALLS   #3,PAS$WRITE_STRING
                 FFFF75D2    EF  9F 014B8         PUSHAB  C.AQZ
                             1D  DD 014BE         PUSHL   #29
                 00000000G   EF  9F 014C0         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 014C6         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 014CD         PUSHAB  CRLF_SHIFT
                             06  DD 014D3         PUSHL   #6
                 00000000G   EF  9F 014D5         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 014DB         CALLS   #3,PAS$WRITE_STRING
                 FFFF75CB    EF  9F 014E2         PUSHAB  C.ARA
                             2B  DD 014E8         PUSHL   #43
                 00000000G   EF  9F 014EA         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 014F0         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 014F7         PUSHAB  CRLF_SHIFT
                             06  DD 014FD         PUSHL   #6
                 00000000G   EF  9F 014FF         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 01505         CALLS   #3,PAS$WRITE_STRING
                 FFFF75CA    EF  9F 0150C         PUSHAB  C.ARB
                             29  DD 01512         PUSHL   #41
                 00000000G   EF  9F 01514         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0151A         CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 01521         PUSHAB  CRLF_SHIFT
                             06  DD 01527         PUSHL   #6
                 00000000G   EF  9F 01529         PUSHAB  PAS$FV_OUTPUT
00000000G  EF                03  FB 0152F         CALLS   #3,PAS$WRITE_STRING
```

Generated Code

```
                FFFF75CC   EF   9F 01536          PUSHAB  C.ARC
                           2A   DD 0153C          PUSHL   #42
                00000000G  EF   9F 0153E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01544          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0154B          PUSHAB  CRLF_SHIFT
                           06   DD 01551          PUSHL   #6
                00000000G  EF   9F 01553          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01559          CALLS   #3,PAS$WRITE_STRING
                FFFF75CE   EF   9F 01560          PUSHAB  C.ARD
                           28   DD 01566          PUSHL   #40
                00000000G  EF   9F 01568          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0156E          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 01575          PUSHAB  CRLF_SHIFT
                           06   DD 0157B          PUSHL   #6
                00000000G  EF   9F 0157D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01583          CALLS   #3,PAS$WRITE_STRING
                FFFF75CC   EF   9F 0158A          PUSHAB  C.ARE
                           29   DD 01590          PUSHL   #41
                00000000G  EF   9F 01592          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01598          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0159F          PUSHAB  CRLF_SHIFT
                           06   DD 015A5          PUSHL   #6
                00000000G  EF   9F 015A7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 015AD          CALLS   #3,PAS$WRITE_STRING
                FFFF75CE   EF   9F 015B4          PUSHAB  C.ARF
                           32   DD 015BA          PUSHL   #50
                00000000G  EF   9F 015BC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 015C2          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 015C9          PUSHAB  CRLF_SHIFT
                           06   DD 015CF          PUSHL   #6
                00000000G  EF   9F 015D1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 015D7          CALLS   #3,PAS$WRITE_STRING
                FFFF75D8   EF   9F 015DE          PUSHAB  C.ARG
                           2A   DD 015E4          PUSHL   #42
                00000000G  EF   9F 015E6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 015EC          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 015F3          PUSHAB  CRLF_SHIFT
                           06   DD 015F9          PUSHL   #6
                00000000G  EF   9F 015FB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01601          CALLS   #3,PAS$WRITE_STRING
                FFFF75DA   EF   9F 01608          PUSHAB  C.ARH
                           29   DD 0160E          PUSHL   #41
                00000000G  EF   9F 01610          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01616          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 0161D          PUSHAB  CRLF_SHIFT
                           06   DD 01623          PUSHL   #6
                00000000G  EF   9F 01625          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 0162B          CALLS   #3,PAS$WRITE_STRING
                FFFF75DC   EF   9F 01632          PUSHAB  C.ARI
                           2E   DD 01638          PUSHL   #46
                00000000G  EF   9F 0163A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01640          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF   9F 01647          PUSHAB  CRLF_SHIFT
                           06   DD 0164D          PUSHL   #6
                00000000G  EF   9F 0164F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF              03   FB 01655          CALLS   #3,PAS$WRITE_STRING
                FFFF75E2   EF   9F 0165C          PUSHAB  C.ARJ
```

```
                                 2E   DD 01662          PUSHL   #46
                    00000000G    EF   9F 01664          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0166A          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 01671          PUSHAB  CRLF_SHIFT
                                 06   DD 01677          PUSHL   #6
                    00000000G    EF   9F 01679          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0167F          CALLS   #3,PAS$WRITE_STRING
                    FFFF75E8     EF   9F 01686          PUSHAB  C.ARK
                                 2E   DD 0168C          PUSHL   #46
                    00000000G    EF   9F 0168E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 01694          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 0169B          PUSHAB  CRLF_SHIFT
                                 06   DD 016A1          PUSHL   #6
                    00000000G    EF   9F 016A3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 016A9          CALLS   #3,PAS$WRITE_STRING
                    FFFF75EE     EF   9F 016B0          PUSHAB  C.ARL
                                 26   DD 016B6          PUSHL   #38
                    00000000G    EF   9F 016B8          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 016BE          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 016C5          PUSHAB  CRLF
                                 02   DD 016CB          PUSHL   #2
                    00000000G    EF   9F 016CD          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 016D3          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 016DA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 016E0          CALLS   #1,PAS$WRITELN2
                               0000V  31 016E7          BRW     64$
                    00000000G    EF   9F 016EA 57$:     PUSHAB  SHIFT                                   ; 5442
                                 04   DD 016F0          PUSHL   #4
                    00000000G    EF   9F 016F2          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 016F8          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 016FF          PUSHAB  QUES_HINT
                                 1F   DD 01705          PUSHL   #31
                    00000000G    EF   9F 01707          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0170D          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 01714          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   01   FB 0171A          CALLS   #1,PAS$WRITELN2
                               0000V  31 01721          BRW     64$
                    00000000     8F   DF 01724 59$:     PUSHAL  #0                                      ; 5450
00000000G   EF                   01   FB 0172A          CALLS   #1,CLEAR
00V00000000G EF                  00   E0 01731          BBS     #0,FULL_PROMPT,61$                      ; 5452
03 00000000G EF                  00   E0 01739          BBS     #0,TEMP_FULL_PROMPT..+3
                               0000V  31 01741          BRW     62$
                    00000000G    EF   9F 01744 61$:     PUSHAB  SHIFT                                   ; 5456
                                 04   DD 0174A          PUSHL   #4
                    00000000G    EF   9F 0174C          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 01752          CALLS   #3,PAS$WRITE_STRING
                    FFFF756B     EF   9F 01759          PUSHAB  C.ARM
                                 02   DD 0175F          PUSHL   #2
                    00000000G    EF   9F 01761          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 01767          CALLS   #3,PAS$WRITE_STRING
                    00000000G    EF   9F 0176E          PUSHAB  ANSI_REVERSE
                                 04   DD 01774          PUSHL   #4
                    00000000G    EF   9F 01776          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                   03   FB 0177C          CALLS   #3,PAS$WRITE_STRING
                    FFFF7543     EF   9F 01783          PUSHAB  C.ARN
                                 0D   DD 01789          PUSHL   #13
                    00000000G    EF   9F 0178B          PUSHAB  PAS$FV_OUTPUT
```

```
                 00000000G  EF                03  FB 01791          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 01798          PUSHAB  SEC_ATTR
                                               16  DD 0179E          PUSHL   #22
                                    00000000G  EF  9F 017A0          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 017A6          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 017AD          PUSHAB  ANSI_RESET
                                               04  DD 017B3          PUSHL   #4
                                    00000000G  EF  9F 017B5          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 017BB          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 017C2          PUSHAB  CRLF
                                               02  DD 017C8          PUSHL   #2
                                    00000000G  EF  9F 017CA          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 017D0          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 017D7          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                01  FB 017DD          CALLS   #1,PAS$WRITELN2
                                    000000FC   8F  DD 017E4          PUSHL   #252                                    ; 5463
                                               07  DD 017EA          PUSHL   #7
                                               04  DD 017EC          PUSHL   #4
                                    00000000G  EF  9F 017EE          PUSHAB  SYS$OUTPUT_NAME
                                               0B  DD 017F4          PUSHL   #11
                                               01  DD 017F6          PUSHL   #1
                                    00000000G  EF  9F 017F8          PUSHAB  FDL_DEST
                 00000000G  EF                07  FB 017FE          CALLS   #7,PAS$OPEN2
                                    00000000G  EF  9F 01805          PUSHAB  FDL_DEST                                ; 5465
                 00000000G  EF                01  FB 0180B          CALLS   #1,PAS$REWRITE2
                                    00000000G  EF  9F 01812          PUSHAB  TEST                                    ; 5467
                 00000000G  EF                01  FB 01818          CALLS   #1,SHOW_PRIMARY_SECTION
                                    00000000G  EF  9F 0181F          PUSHAB  FDL_DEST                                ; 5469
                 00000000G  EF                01  FB 01825          CALLS   #1,PAS$CLOSE2
                                                00V 11  0182C          BRB     63$
                                    00000000G  EF  9F 0182E 62$:     PUSHAB  SHIFT                                   ; 5475
                                               04  DD 01834          PUSHL   #4
                                    00000000G  EF  9F 01836          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 0183C          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 01843          PUSHAB  QUES_HINT
                                               1F  DD 01849          PUSHL   #31
                                    00000000G  EF  9F 0184B          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 01851          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 01858          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                01  FB 0185E          CALLS   #1,PAS$WRITELN2
                                                    01865 63$:
                                    00000000G  EF  9F 01865 64$:     PUSHAB  SHIFT                                   ; 5479
                                               04  DD 0186B          PUSHL   #4
                                    00000000G  EF  9F 0186D          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 01873          CALLS   #3,PAS$WRITE_STRING
                                    FFFF745C   EF  9F 0187A          PUSHAB  C_ARO
                                               1F  DD 01880          PUSHL   #31
                                    00000000G  EF  9F 01882          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 01888          CALLS   #3,PAS$WRITE_STRING
                                    00000000G  EF  9F 0188F          PUSHAB  ANSI_REVERSE
                                               04  DD 01895          PUSHL   #4
                                    00000000G  EF  9F 01897          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 0189D          CALLS   #3,PAS$WRITE_STRING
                                    FFFF7452   EF  9F 018A4          PUSHAB  C_ARP
                                               03  DD 018AA          PUSHL   #3
                                    00000000G  EF  9F 018AC          PUSHAB  PAS$FV_OUTPUT
                 00000000G  EF                03  FB 018B2          CALLS   #3,PAS$WRITE_STRING
```

```
                    00000000G  EF  9F 018B9          PUSHAB  ANSI_RESET
                               04  DD 018BF          PUSHL   #4
                    00000000G  EF  9F 018C1          PUSHAB  PAS$FV_OUTPUT
    00000000G  EF              03  FB 018C7          CALLS   #3,PAS$WRITE_STRING
                    FFFF742C   EF  9F 018CE          PUSHAB  C.ARQ
                               03  DD 018D4          PUSHL   #3
                    00000000G  EF  9F 018D6          PUSHAB  PAS$FV_OUTPUT
    00000000G  EF              03  FB 018DC          CALLS   #3,PAS$WRITE_STRING
                    00000000   8F  DF 018E3          PUSHAL  #0                                      ; 5481
                          00   8F  9F 018E9          PUSHAB  #0
          F4   AD 00000000G    EF  9E 018EC          MOVAB   EDF$AB_FILE_TABLE_STA,-12(FP)
                          F4   AD  9F 018F4          PUSHAB  -12(FP)
          F0   AD 00000000G    EF  9E 018F7          MOVAB   EDF$AB_FILE_TABLE_KEY,-16(FP)
                          F0   AD  9F 018FF          PUSHAB  -16(FP)
    00000000G  EF              04  FB 01902          CALLS   #4,PARSE_INPUT
                         0000V 31  01909          BRW     126$
 03 00000000G  EF              00  E0 0190C 65$:     BBS     #0,FULL_CHOICE,.+3                      ; 5494
                         0000V 31  01914          BRW     71$
                    00000000   8F  DF 01917          PUSHAL  #0                                      ; 5498
    00000000G  EF              01  FB 0191D          CALLS   #1,CLEAR
00V00000000G  EF              00  E0 01924          BBS     #0,FULL_PROMPT,68$                      ; 5500
 03 00000000G  EF              00  E0 0192C          BBS     #0,TEMP_FULL_PROMPT,.+3
                         0000V 31  01934          BRW     69$
                    00000000G  EF  9F 01937 68$:     PUSHAB  SHIFT                                   ; 5504
                               04  DD 0193D          PUSHL   #4
                    00000000G  EF  9F 0193F          PUSHAB  PAS$FV_OUTPUT
    00000000G  EF              03  FB 01945          CALLS   #3,PAS$WRITE_STRING
                    FFFF73B2   EF  9F 0194C          PUSHAB  C.ARR
                               02  DD 01952          PUSHL   #2
                    00000000G  EF  9F 01954          PUSHAB  PAS$FV_OUTPUT
    00000000G  EF              03  FB 0195A          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 01961          PUSHAB  ANSI_REVERSE
                               04  DD 01967          PUSHL   #4
                    00000000G  EF  9F 01969          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 0196F          CALLS   #3,PAS$WRITE_STRING
                    FFFF738C   EF  9F 01976          PUSHAB  C.ARS
                               0E  DD 0197C          PUSHL   #14
                    00000000G  EF  9F 0197E          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 01984          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 0198B          PUSHAB  SEC_ATTR
                               16  DD 01991          PUSHL   #22
                    00000000G  EF  9F 01993          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 01999          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 019A0          PUSHAB  ANSI_RESET
                               04  DD 019A6          PUSHL   #4
                    00000000G  EF  9F 019A8          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 019AE          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 019B5          PUSHAB  CRLF
                               02  DD 019BB          PUSHL   #2
                    00000000G  EF  9F 019BD          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 019C3          CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF  9F 019CA          PUSHAB  CRLF_SHIFT
                               06  DD 019D0          PUSHL   #6
                    00000000G  EF  9F 019D2          PUSHAB  PAS$FV_OUTPUT
 00000000G  EF              03  FB 019D8          CALLS   #3,PAS$WRITE_STRING
                    FFFF7333   EF  9F 019DF          PUSHAB  C.ART
                               13  DD 019E5          PUSHL   #19
```

```
                                Generated Code

                         00000000G   EF   9F 019E7         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 019ED         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 019F4         PUSHAB  CRLF_SHIFT
                                     06   DD 019FA         PUSHL   #6
                         00000000G   EF   9F 019FC         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A02         CALLS   #3,PAS$WRITE_STRING
                         FFFF731D    EF   9F 01A09         PUSHAB  C.ARU
                                     12   DD 01A0F         PUSHL   #18
                         00000000G   EF   9F 01A11         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A17         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01A1E         PUSHAB  CRLF_SHIFT
                                     06   DD 01A24         PUSHL   #6
                         00000000G   EF   9F 01A26         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A2C         CALLS   #3,PAS$WRITE_STRING
                         FFFF7307    EF   9F 01A33         PUSHAB  C.ARV
                                     13   DD 01A39         PUSHL   #19
                         00000000G   EF   9F 01A3B         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A41         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01A48         PUSHAB  CRLF_SHIFT
                                     06   DD 01A4E         PUSHL   #6
                         00000000G   EF   9F 01A50         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A56         CALLS   #3,PAS$WRITE_STRING
                         FFFF72F1    EF   9F 01A5D         PUSHAB  C.ARW
                                     12   DD 01A63         PUSHL   #18
                         00000000G   EF   9F 01A65         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A6B         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01A72         PUSHAB  CRLF_SHIFT
                                     06   DD 01A78         PUSHL   #6
                         00000000G   EF   9F 01A7A         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A80         CALLS   #3,PAS$WRITE_STRING
                         FFFF72DB    EF   9F 01A87         PUSHAB  C.ARX
                                     14   DD 01A8D         PUSHL   #20
                         00000000G   EF   9F 01A8F         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01A95         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01A9C         PUSHAB  CRLF_SHIFT
                                     06   DD 01AA2         PUSHL   #6
                         00000000G   EF   9F 01AA4         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01AAA         CALLS   #3,PAS$WRITE_STRING
                         FFFF72C5    EF   9F 01AB1         PUSHAB  C.ARY
                                     13   DD 01AB7         PUSHL   #19
                         00000000G   EF   9F 01AB9         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01ABF         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01AC6         PUSHAB  CRLF_SHIFT
                                     06   DD 01ACC         PUSHL   #6
                         00000000G   EF   9F 01ACE         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01AD4         CALLS   #3,PAS$WRITE_STRING
                         FFFF72AF    EF   9F 01ADB         PUSHAB  C.ARZ
                                     16   DD 01AE1         PUSHL   #22
                         00000000G   EF   9F 01AE3         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01AE9         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01AF0         PUSHAB  CRLF
                                     02   DD 01AF6         PUSHL   #2
                         00000000G   EF   9F 01AF8         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       03   FB 01AFE         CALLS   #3,PAS$WRITE_STRING
                         00000000G   EF   9F 01B05         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                       01   FB 01B0B         CALLS   #1,PAS$WRITELN2
                                  0000V   31   01B12         BRW     76$
```

```
                    00000000G   EF    9F 01B15 69$:    PUSHAB  SHIFT                                   ; 5529
                                04    DD 01B1B         PUSHL   #4
                    00000000G   EF    9F 01B1D         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01B23         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01B2A         PUSHAB  QUES_HINT
                                1F    DD 01B30         PUSHL   #31
                    00000000G   EF    9F 01B32         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01B38         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01B3F         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              01    FB 01B45         CALLS   #1,PAS$WRITELN2
                              0000V   31 01B4C         BRW     76$
                    00000000    8F    DF 01B4F 71$:    PUSHAL  #0                                      ; 5539
    00000000G   EF              01    FB 01B55         CALLS   #1,CLEAR
00V00000000G    EF              00    E0 01B5C         BBS     #0,FULL_PROMPT,73$                      ; 5541
03  00000000G   EF              00    E0 01B64         BBS     #0,TEMP_FULL_PROMPT,.+3
                              0000V   31 01B6C         BRW     74$
                    00000000G   EF    9F 01B6F 73$:    PUSHAB  SHIFT                                   ; 5545
                                04    DD 01B75         PUSHL   #4
                    00000000G   EF    9F 01B77         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01B7D         CALLS   #3,PAS$WRITE_STRING
                    FFFF721C    EF    9F 01B84         PUSHAB  C.ASA
                                02    DD 01B8A         PUSHL   #2
                    00000000G   EF    9F 01B8C         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01B92         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01B99         PUSHAB  ANSI_REVERSE
                                04    DD 01B9F         PUSHL   #4
                    00000000G   EF    9F 01BA1         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01BA7         CALLS   #3,PAS$WRITE_STRING
                    FFFF71F4    EF    9F 01BAE         PUSHAB  C.ASB
                                10    DD 01BB4         PUSHL   #16
                    00000000G   EF    9F 01BB6         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01BBC         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01BC3         PUSHAB  SEC_ATTR
                                16    DD 01BC9         PUSHL   #22
                    00000000G   EF    9F 01BCB         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01BD1         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01BD8         PUSHAB  ANSI_RESET
                                04    DD 01BDE         PUSHL   #4
                    00000000G   EF    9F 01BE0         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01BE6         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01BED         PUSHAB  CRLF
                                02    DD 01BF3         PUSHL   #2
                    00000000G   EF    9F 01BF5         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              03    FB 01BFB         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF    9F 01C02         PUSHAB  PAS$FV_OUTPUT
    00000000G   EF              01    FB 01C08         CALLS   #1,PAS$WRITELN2
                    000000FC    8F    DD 01C0F         PUSHL   #252                                    ; 5552
                                07    DD 01C15         PUSHL   #7
                                04    DD 01C17         PUSHL   #4
                    00000000G   EF    9F 01C19         PUSHAB  SYS$OUTPUT_NAME
                                0B    DD 01C1F         PUSHL   #11
                                01    DD 01C21         PUSHL   #1
                    00000000G   EF    9F 01C23         PUSHAB  FDL_DEST
    00000000G   EF              07    FB 01C29         CALLS   #7,PAS$OPEN2
                    00000000G   EF    9F 01C30         PUSHAB  FDL_DEST                                ; 5554
    00000000G   EF              01    FB 01C36         CALLS   #1,PAS$REWRITE2
                    00000000G   EF    9F 01C3D         PUSHAB  TEST                                    ; 5556
```

```
                                    01 FB 01C43          CALLS   #1,SHOW_PRIMARY_SECTION
                00000000G  EF       9F 01C4A          PUSHAB  FDL_DEST                          ; 5558
00000000G  EF                       01 FB 01C50          CALLS   #1,PAS$CLOSE2
                                    00V 11 01C57         BRB     76$
                00000000G  EF       9F 01C59 74$:     PUSHAB  SHIFT                            ; 5564
                           04       DD 01C5F          PUSHL   #4
                00000000G  EF       9F 01C61          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01C67          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF       9F 01C6E          PUSHAB  QUES_HINT
                           1F       DD 01C74          PUSHL   #31
                00000000G  EF       9F 01C76          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01C7C          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF       9F 01C83          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       01 FB 01C89          CALLS   #1,PAS$WRITELN2
                00000000G  EF       9F 01C90 76$:     PUSHAB  SHIFT                            ; 5568
                           04       DD 01C96          PUSHL   #4
                00000000G  EF       9F 01C98          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01C9E          CALLS   #3,PAS$WRITE_STRING
                FFFF710D   EF       9F 01CA5          PUSHAB  C.ASC
                           22       DD 01CAB          PUSHL   #34
                00000000G  EF       9F 01CAD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01CB3          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF       9F 01CBA          PUSHAB  ANSI_REVERSE
                           04       DD 01CC0          PUSHL   #4
                00000000G  EF       9F 01CC2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01CC8          CALLS   #3,PAS$WRITE_STRING
                FFFF7107   EF       9F 01CCF          PUSHAB  C.ASD
                           03       DD 01CD5          PUSHL   #3
                00000000G  EF       9F 01CD7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03 FB 01CDD          CALLS   #3,PAS$WRITE_STRING
                00000000G  EF       9F 01CE4          PUSHAB  ANSI_RESET
                           04       DD 01CEA          PUSHL   #4
                00000000G  EF       9F 01CEC          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                    03 FB 01CF2          CALLS   #3,PAS$WRITE_STRING
                FFFF70E1   EF       9F 01CF9          PUSHAB  C.ASE
                           03       DD 01CFF          PUSHL   #3
                00000000G  EF       9F 01D01          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                    03 FB 01D07          CALLS   #3,PAS$WRITE_STRING
                00000000   8F       DF 01D0E          PUSHAL  #0                               ; 5570
                      00   8F       9F 01D14          PUSHAB  #0
         F4    AD 00000000G  EF     9E 01D17          MOVAB   EDF$AB_JOURNAL_TABLE_STA,-12(FP)
                      F4    AD     9F 01D1F          PUSHAB  -12(FP)
         F0    AD 00000000G  EF     9E 01D22          MOVAB   EDF$AB_JOURNAL_TABLE_KEY,-16(FP)
                      F0    AD     9F 01D2A          PUSHAB  -16(FP)
   00000000G  EF                    04 FB 01D2D          CALLS   #4,PARSE_INPUT
                           0000V    31 01D34          BRW     126$
03 00000000G  EF                    00 E0 01D37 77$:     BBS     #0,FULL_CHOICE,..+3          ; 5583
                           0000V    31 01D3F          BRW     83$
                00000000   8F       DF 01D42          PUSHAL  #0                               ; 5587
   00000000G  EF                    01 FB 01D48          CALLS   #1,CLEAR
00V00000000G  EF                    00 E0 01D4F          BBS     #0,FULL_PROMPT,80$           ; 5589
03 00000000G  EF                    00 E0 01D57          BBS     #0,TEMP_FULL_PROMPT,..+3
                           0000V    31 01D5F          BRW     81$
                00000000G  EF       9F 01D62 80$:     PUSHAB  SHIFT                            ; 5593
                           04       DD 01D68          PUSHL   #4
                00000000G  EF       9F 01D6A          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF                    03 FB 01D70          CALLS   #3,PAS$WRITE_STRING
```

```
                 FFFF7067   EF   9F 01D77         PUSHAB  C.ASF
                            02   DD 01D7D         PUSHL   #2
                 00000000G  EF   9F 01D7F         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01D85         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01D8C         PUSHAB  ANSI_REVERSE
                            04   DD 01D92         PUSHL   #4
                 00000000G  EF   9F 01D94         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01D9A         CALLS   #3,PAS$WRITE_STRING
                 FFFF7041   EF   9F 01DA1         PUSHAB  C.ASG
                            0A   DD 01DA7         PUSHL   #10
                 00000000G  EF   9F 01DA9         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01DAF         CALLS   #3,PAS$WRITE_STRING
                            03   DD 01DB6         PUSHL   #3
                 00000084G  EF   DD 01DB8         PUSHL   IDATA+132
                 00000000G  EF   9F 01DBE         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01DC4         CALLS   #3,PAS$WRITE_INTEGER
                 00000000G  EF   9F 01DCB         PUSHAB  SEC_ATTR
                            16   DD 01DD1         PUSHL   #22
                 00000000G  EF   9F 01DD3         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01DD9         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01DE0         PUSHAB  ANSI_RESET
                            04   DD 01DE6         PUSHL   #4
                 00000000G  EF   9F 01DE8         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01DEE         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01DF5         PUSHAB  CRLF
                            02   DD 01DFB         PUSHL   #2
                 00000000G  EF   9F 01DFD         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E03         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01E0A         PUSHAB  CRLF_SHIFT
                            06   DD 01E10         PUSHL   #6
                 00000000G  EF   9F 01E12         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E18         CALLS   #3,PAS$WRITE_STRING
                 FFFF6FCF   EF   9F 01E1F         PUSHAB  C.ASH
                            29   DD 01E25         PUSHL   #41
                 00000000G  EF   9F 01E27         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E2D         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01E34         PUSHAB  CRLF_SHIFT
                            06   DD 01E3A         PUSHL   #6
                 00000000G  EF   9F 01E3C         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E42         CALLS   #3,PAS$WRITE_STRING
                 FFFF6FD1   EF   9F 01E49         PUSHAB  C.ASI
                            1F   DD 01E4F         PUSHL   #31
                 00000000G  EF   9F 01E51         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E57         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01E5E         PUSHAB  CRLF_SHIFT
                            06   DD 01E64         PUSHL   #6
                 00000000G  EF   9F 01E66         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E6C         CALLS   #3,PAS$WRITE_STRING
                 FFFF6FC7   EF   9F 01E73         PUSHAB  C.ASJ
                            22   DD 01E79         PUSHL   #34
                 00000000G  EF   9F 01E7B         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E81         CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 01E88         PUSHAB  CRLF_SHIFT
                            06   DD 01E8E         PUSHL   #6
                 00000000G  EF   9F 01E90         PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 01E96         CALLS   #3,PAS$WRITE_STRING
                 FFFF6FC1   EF   9F 01E9D         PUSHAB  C.ASK
```

Generated Code

```
                                30   DD 01EA3         PUSHL   #48
                    00000000G   EF   9F 01EA5         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01EAB         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01EB2         PUSHAB  CRLF_SHIFT
                                06   DD 01EB8         PUSHL   #6
                    00000000G   EF   9F 01EBA         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01EC0         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FC7    EF   9F 01EC7         PUSHAB  C.ASL
                                2F   DD 01ECD         PUSHL   #47
                    00000000G   EF   9F 01ECF         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01ED5         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01EDC         PUSHAB  CRLF_SHIFT
                                06   DD 01EE2         PUSHL   #6
                    00000000G   EF   9F 01EE4         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01EEA         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FCD    EF   9F 01EF1         PUSHAB  C.ASM
                                22   DD 01EF7         PUSHL   #34
                    00000000G   EF   9F 01EF9         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01EFF         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01F06         PUSHAB  CRLF_SHIFT
                                06   DD 01F0C         PUSHL   #6
                    00000000G   EF   9F 01F0E         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F14         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FC7    EF   9F 01F1B         PUSHAB  C.ASN
                                21   DD 01F21         PUSHL   #33
                    00000000G   EF   9F 01F23         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F29         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01F30         PUSHAB  CRLF_SHIFT
                                06   DD 01F36         PUSHL   #6
                    00000000G   EF   9F 01F38         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F3E         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FC1    EF   9F 01F45         PUSHAB  C.ASO
                                2C   DD 01F4B         PUSHL   #44
                    00000000G   EF   9F 01F4D         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F53         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01F5A         PUSHAB  CRLF_SHIFT
                                06   DD 01F60         PUSHL   #6
                    00000000G   EF   9F 01F62         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F68         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FC3    EF   9F 01F6F         PUSHAB  C.ASP
                                28   DD 01F75         PUSHL   #40
                    00000000G   EF   9F 01F77         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F7D         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01F84         PUSHAB  CRLF_SHIFT
                                06   DD 01F8A         PUSHL   #6
                    00000000G   EF   9F 01F8C         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01F92         CALLS   #3,PAS$WRITE_STRING
                    FFFF6FC1    EF   9F 01F99         PUSHAB  C.ASQ
                                0F   DD 01F9F         PUSHL   #15
                    00000000G   EF   9F 01FA1         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01FA7         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01FAE         PUSHAB  CRLF
                                02   DD 01FB4         PUSHL   #2
                    00000000G   EF   9F 01FB6         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  03   FB 01FBC         CALLS   #3,PAS$WRITE_STRING
                    00000000G   EF   9F 01FC3         PUSHAB  PAS$FV_OUTPUT
00000000G   EF                  01   FB 01FC9         CALLS   #1,PAS$WRITELN2
```

```
                                00V  11 01FD0          BRB     82$
                         00000000G  EF 9F 01FD2 81$:   PUSHAB  SHIFT                                  ; 5624
                                    04 DD 01FD8         PUSHL   #4
                         00000000G  EF 9F 01FDA         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 01FE0         CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF 9F 01FE7         PUSHAB  QUES_HINT
                                    1F DD 01FED         PUSHL   #31
                         00000000G  EF 9F 01FEF         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 01FF5         CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF 9F 01FFC         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    01 FB 02002         CALLS   #1,PAS$WRITELN2
                              0000V 31 02009 82$:       BRW     88$
                         00000000   8F DF 0200C 83$:    PUSHAL  #0                                   ; 5632
                   00000000G  EF    01 FB 02012         CALLS   #1,CLEAR
                00V00000000G  EF    00 E0 02019         BBS     #0,FULL_PROMPT,85$                   ; 5634
                03 00000000G  EF    00 E0 02021         BBS     #0,TEMP_FULL_PROMPT,.+3
                              0000V 31 02029         BRW     86$
                         00000000G  EF 9F 0202C 85$:    PUSHAB  SHIFT                                ; 5638
                                    04 DD 02032         PUSHL   #4
                         00000000G  EF 9F 02034         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 0203A         CALLS   #3,PAS$WRITE_STRING
                         FFFF6F29   EF 9F 02041         PUSHAB  C.ASR
                                    02 DD 02047         PUSHL   #2
                         00000000G  EF 9F 02049         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 0204F         CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF 9F 02056         PUSHAB  ANSI_REVERSE
                                    04 DD 0205C         PUSHL   #4
                         00000000G  EF 9F 0205E         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 02064         CALLS   #3,PAS$WRITE_STRING
                         FFFF6F03   EF 9F 0206B         PUSHAB  C.ASS
                                    0C DD 02071         PUSHL   #12
                         00000000G  EF 9F 02073         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 02079         CALLS   #3,PAS$WRITE_STRING
                                    03 DD 02080         PUSHL   #3
                         00000084G  EF DD 02082         PUSHL   IDATA+132
                         00000000G  EF 9F 02088         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 0208E         CALLS   #3,PAS$WRITE_INTEGER
                         00000000G  EF 9F 02095         PUSHAB  SEC_ATTR
                                    16 DD 0209B         PUSHL   #22
                         00000000G  EF 9F 0209D         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 020A3         CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF 9F 020AA         PUSHAB  ANSI_RESET
                                    04 DD 020B0         PUSHL   #4
                         00000000G  EF 9F 020B2         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 020B8         CALLS   #3,PAS$WRITE_STRING
                         000000C0G  EF 9F 020BF         PUSHAB  CRLF
                                    02 DD 020C5         PUSHL   #2
                         00000000G  EF 9F 020C7         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    03 FB 020CD         CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF 9F 020D4         PUSHAB  PAS$FV_OUTPUT
                   00000000G  EF    01 FB 020DA         CALLS   #1,PAS$WRITELN2
                         000000FC   8F DD 020E1         PUSHL   #252                                 ; 5647
                                    07 DD 020E7         PUSHL   #7
                                    04 DD 020E9         PUSHL   #4
                         00000000G  EF 9F 020EB         PUSHAB  SYS$OUTPUT_NAME
                                    0B DD 020F1         PUSHL   #11
                                    01 DD 020F3         PUSHL   #1
```

```
                 00000000G  EF  9F 020F5          PUSHAB  FDL_DEST
00000000G  EF    07         FB 020FB          CALLS   #7,PAS$OPEN2
                 00000000G  EF  9F 02102          PUSHAB  FDL_DEST                                ; 5649
00000000G  EF    01         FB 02108          CALLS   #1,PAS$REWRITE2
                 00000000G  EF  9F 0210F          PUSHAB  TEST                                    ; 5651
00000000G  EF    01         FB 02115          CALLS   #1,SHOW_PRIMARY_SECTION
                 00000000G  EF  9F 0211C          PUSHAB  FDL_DEST                                ; 5653
00000000G  EF    01         FB 02122          CALLS   #1,PAS$CLOSE2
                 00V        11 02129          BRB     88$
                 00000000G  EF  9F 0212B 86$:     PUSHAB  SHIFT                                   ; 5659
                 04         DD 02131          PUSHL   #4
                 00000000G  EF  9F 02133          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 02139          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 02140          PUSHAB  QUES_HINT
                 1F         DD 02146          PUSHL   #31
                 00000000G  EF  9F 02148          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 0214E          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 02155          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    01         FB 0215B          CALLS   #1,PAS$WRITELN2
                 00000000G  EF  9F 02162 88$:     PUSHAB  SHIFT                                   ; 5663
                 04         DD 02168          PUSHL   #4
                 00000000G  EF  9F 0216A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 02170          CALLS   #3,PAS$WRITE_STRING
                 FFFF6E03   EF  9F 02177          PUSHAB  C.AST
                 09         DD 0217D          PUSHL   #9
                 00000000G  EF  9F 0217F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 02185          CALLS   #3,PAS$WRITE_STRING
                 03         DD 0218C          PUSHL   #3
                 00000084G  EF  DD 0218E          PUSHL   IDATA+132
                 00000000G  EF  9F 02194          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 0219A          CALLS   #3,PAS$WRITE_INTEGER
                 FFFF6DE5   EF  9F 021A1          PUSHAB  C.ASU
                 15         DD 021A7          PUSHL   #21
                 00000000G  EF  9F 021A9          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 021AF          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 021B6          PUSHAB  ANSI_REVERSE
                 04         DD 021BC          PUSHL   #4
                 00000000G  EF  9F 021BE          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 021C4          CALLS   #3,PAS$WRITE_STRING
                 FFFF6DD3   EF  9F 021CB          PUSHAB  C.ASV
                 03         DD 021D1          PUSHL   #3
                 00000000G  EF  9F 021D3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 021D9          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F 021E0          PUSHAB  ANSI_RESET
                 04         DD 021E6          PUSHL   #4
                 00000000G  EF  9F 021E8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 021EE          CALLS   #3,PAS$WRITE_STRING
                 FFFF6DAD   EF  9F 021F5          PUSHAB  C.ASW
                 03         DD 021FB          PUSHL   #3
                 00000000G  EF  9F 021FD          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB 02203          CALLS   #3,PAS$WRITE_STRING
                 00000000   8F  DF 0220A          PUSHAL  #0                                      ; 5667
                       00   8F  9F 02210          PUSHAB  #0
      F4   AD 00000000G  EF  9E 02213          MOVAB   EDF$AB_KEY_TABLE_STA,-12(FP)
                       F4   AD  9F 0221B          PUSHAB  -12(FP)
      F0   AD 00000000G  EF  9E 0221E          MOVAB   EDF$AB_KEY_TABLE_KEY,-16(FP)
                       F0   AD  9F 02226          PUSHAB  -16(FP)
```

```
                 00000000G  EF              04   FB 02229          CALLS   #4,PARSE_INPUT
                                        0000V 31   02230          BRW     126$
03 00000000G  EF                 00     E0   02233 89$:         BBS     #0,FULL_CHOICE,..+3                     ; 5680
                                        0000V 31   0223B          BRW     95$
                          00000000       8F   DF   0223E          PUSHAL  #0                                      ; 5684
                 00000000G  EF            01   FB   02244          CALLS   #1,CLEAR
00V00000000G  EF                 00     E0   0224B          BBS     #0,FULL_PROMPT,92$                      ; 5686
03 00000000G  EF                 00     E0   02253          BBS     #0,TEMP_FULL_PROMPT,..+3
                                        0000V 31   0225B          BRW     93$
                             00000000G   EF   9F   0225E 92$:      PUSHAB  SHIFT                                   ; 5690
                                         04   DD   02264          PUSHL   #4
                             00000000G   EF   9F   02266          PUSHAB  PAS$FV_OUTPUT
                 00000000G   EF               03   FB   0226C          CALLS   #3,PAS$WRITE_STRING
                             FFFF6D33    EF   9F   02273          PUSHAB  C.ASX
                                         02   DD   02279          PUSHL   #2
                             00000000G   EF   9F   0227B          PUSHAB  PAS$FV_OUTPUT
                 00000000G   EF               03   FB   02281          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   02288          PUSHAB  ANSI_REVERSE
                                         04   DD   0228E          PUSHL   #4
                             00000000G   EF   9F   02290          PUSHAB  PAS$FV_OUTPUT
                 00000000G   EF               03   FB   02296          CALLS   #3,PAS$WRITE_STRING
                             FFFF6D0D    EF   9F   0229D          PUSHAB  C.ASY
                                         0D   DD   022A3          PUSHL   #13
                             00000000G   EF   9F   022A5          PUSHAB  PAS$FV_OUTPUT
                 00000000G   EF               03   FB   022AB          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   022B2          PUSHAB  SEC_ATTR
                                         16   DD   022B8          PUSHL   #22
                             00000000G   EF   9F   022BA          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   022C0          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   022C7          PUSHAB  ANSI_RESET
                                         04   DD   022CD          PUSHL   #4
                             00000000G   EF   9F   022CF          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   022D5          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   022DC          PUSHAB  CRLF
                                         02   DD   022E2          PUSHL   #2
                             00000000G   EF   9F   022E4          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   022EA          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   022F1          PUSHAB  CRLF_SHIFT
                                         06   DD   022F7          PUSHL   #6
                             00000000G   EF   9F   022F9          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   022FF          CALLS   #3,PAS$WRITE_STRING
                             FFFF6CB4    EF   9F   02306          PUSHAB  C.ASZ
                                         12   DD   0230C          PUSHL   #18
                             00000000G   EF   9F   0230E          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   02314          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   0231B          PUSHAB  CRLF_SHIFT
                                         06   DD   02321          PUSHL   #6
                             00000000G   EF   9F   02323          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   02329          CALLS   #3,PAS$WRITE_STRING
                             FFFF6C9E    EF   9F   02330          PUSHAB  C.ATA
                                         18   DD   02336          PUSHL   #24
                             00000000G   EF   9F   02338          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   0233E          CALLS   #3,PAS$WRITE_STRING
                             00000000G   EF   9F   02345          PUSHAB  CRLF_SHIFT
                                         06   DD   0234B          PUSHL   #6
                             00000000G   EF   9F   0234D          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                           03   FB   02353          CALLS   #3,PAS$WRITE_STRING
```

```
                    FFFF6C8C   EF   9F 0235A          PUSHAB  C.ATB
                               19   DD 02360          PUSHL   #25
                    00000000G  EF   9F 02362          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 02368      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0236F          PUSHAB  CRLF_SHIFT
                               06   DD 02375          PUSHL   #6
                    00000000G  EF   9F 02377          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 0237D      CALLS   #3,PAS$WRITE_STRING
                    FFFF6C7E   EF   9F 02384          PUSHAB  C.ATC
                               10   DD 0238A          PUSHL   #16
                    00000000G  EF   9F 0238C          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 02392      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 02399          PUSHAB  CRLF_SHIFT
                               06   DD 0239F          PUSHL   #6
                    00000000G  EF   9F 023A1          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 023A7      CALLS   #3,PAS$WRITE_STRING
                    FFFF6C64   EF   9F 023AE          PUSHAB  C.ATD
                               0D   DD 023B4          PUSHL   #13
                    00000000G  EF   9F 023B6          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 023BC      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 023C3          PUSHAB  CRLF
                               02   DD 023C9          PUSHL   #2
                    00000000G  EF   9F 023CB          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 023D1      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 023D8          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               01 FB 023DE      CALLS   #1,PAS$WRITELN2
                              0000V  31   023E5        BRW     100$
                    00000000G  EF   9F 023E8 93$:     PUSHAB  SHIFT                                   ; 5711
                               04   DD 023EE          PUSHL   #4
                    00000000G  EF   9F 023F0          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 023F6      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 023FD          PUSHAB  QUES_HINT
                               1F   DD 02403          PUSHL   #31
                    00000000G  EF   9F 02405          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 0240B      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 02412          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               01 FB 02418      CALLS   #1,PAS$WRITELN2
                              0000V  31   0241F        BRW     100$
                    00000000   8F   DF 02422 95$:     PUSHAL  #0                                      ; 5721
          00000000G EF               01 FB 02428      CALLS   #1,CLEAR
    00V00000000G    EF               00 E0 0242F      BBS     #0,FULL_PROMPT,97$                      ; 5723
03  00000000G       EF               00 E0 02437      BBS     #0,TEMP_FULL_PROMPT,..+3
                              0000V  31   0243F        BRW     98$
                    00000000G  EF   9F 02442 97$:     PUSHAB  SHIFT                                   ; 5727
                               04   DD 02448          PUSHL   #4
                    00000000G  EF   9F 0244A          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 02450      CALLS   #3,PAS$WRITE_STRING
                    FFFF6BC9   EF   9F 02457          PUSHAB  C.ATE
                               02   DD 0245D          PUSHL   #2
                    00000000G  EF   9F 0245F          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 02465      CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0246C          PUSHAB  ANSI_REVERSE
                               04   DD 02472          PUSHL   #4
                    00000000G  EF   9F 02474          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03 FB 0247A      CALLS   #3,PAS$WRITE_STRING
                    FFFF6BA1   EF   9F 02481          PUSHAB  C.ATF
                               0F   DD 02487          PUSHL   #15
```

```
                 00000000G  EF  9F  02489          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  0248F            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02496          PUSHAB  SEC_ATTR
                             16  DD  0249C          PUSHL   #22
                 00000000G  EF  9F  0249E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  024A4            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  024AB          PUSHAB  ANSI_RESET
                             04  DD  024B1          PUSHL   #4
                 00000000G  EF  9F  024B3          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  024B9            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  024C0          PUSHAB  CRLF
                             02  DD  024C6          PUSHL   #2
                 00000000G  EF  9F  024C8          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  024CE            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  024D5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    01         FB  024DB            CALLS   #1,PAS$WRITELN2
                 000000FC   8F  DD  024E2          PUSHL   #252                        ; 5734
                             07  DD  024E8          PUSHL   #7
                             04  DD  024EA          PUSHL   #4
                 00000000G  EF  9F  024EC          PUSHAB  SYS$OUTPUT_NAME
                             0B  DD  024F2          PUSHL   #11
                             01  DD  024F4          PUSHL   #1
                 00000000G  EF  9F  024F6          PUSHAB  FDL_DEST
00000000G  EF    07         FB  024FC            CALLS   #7,PAS$OPEN2
                 00000000G  EF  9F  02503          PUSHAB  FDL_DEST                    ; 5736
00000000G  EF    01         FB  02509            CALLS   #1,PAS$REWRITE2
                 00000000G  EF  9F  02510          PUSHAB  TEST                        ; 5738
00000000G  EF    01         FB  02516            CALLS   #1,SHOW_PRIMARY_SECTION
                 00000000G  EF  9F  0251D          PUSHAB  FDL_DEST                    ; 5740
00000000G  EF    01         FB  02523            CALLS   #1,PAS$CLOSE2
                             00V 11  0252A          BRB     100$
                 00000000G  EF  9F  0252C  98$:    PUSHAB  SHIFT                       ; 5746
                             04  DD  02532          PUSHL   #4
                 00000000G  EF  9F  02534          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  0253A            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02541          PUSHAB  QUES_HINT
                             1F  DD  02547          PUSHL   #31
                 00000000G  EF  9F  02549          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  0254F            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  02556          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    01         FB  0255C            CALLS   #1,PAS$WRITELN2
                 00000000G  EF  9F  02563  100$:   PUSHAB  SHIFT                       ; 5750
                             04  DD  02569          PUSHL   #4
                 00000000G  EF  9F  0256B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  02571            CALLS   #3,PAS$WRITE_STRING
                 FFFF6ABA   EF  9F  02578          PUSHAB  C_ATG
                             21  DD  0257E          PUSHL   #33
                 00000000G  EF  9F  02580          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  02586            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF  9F  0258D          PUSHAB  ANSI_REVERSE
                             04  DD  02593          PUSHL   #4
                 00000000G  EF  9F  02595          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  0259B            CALLS   #3,PAS$WRITE_STRING
                 FFFF6AB4   EF  9F  025A2          PUSHAB  C_ATH
                             03  DD  025A8          PUSHL   #3
                 00000000G  EF  9F  025AA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF    03         FB  025B0            CALLS   #3,PAS$WRITE_STRING
```

```
                    00000000G  EF    9F 025B7          PUSHAB  ANSI_RESET
                               04    DD 025BD          PUSHL   #4
                    00000000G  EF    9F 025BF          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03    FB 025C5    CALLS   #3,PAS$WRITE_STRING
                    FFFF6A8E   EF    9F 025CC          PUSHAB  C.ATI
                               03    DD 025D2          PUSHL   #3
                    00000000G  EF    9F 025D4          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03    FB 025DA    CALLS   #3,PAS$WRITE_STRING
                    00000000   8F    DF 025E1          PUSHAL  #0                                      ; 5752
                          00   8F    9F 025E7          PUSHAB  #0
                F4    AD 00000000G  EF    9E 025EA          MOVAB   EDF$AB_RECORD_TABLE_STA,-12(FP)
                         F4    AD    9F 025F2          PUSHAB  -12(FP)
                F0    AD 00000000G  EF    9E 025F5          MOVAB   EDF$AB_RECORD_TABLE_KEY,-16(FP)
                         F0    AD    9F 025FD          PUSHAB  -16(FP)
          00000000G EF               04    FB 02600    CALLS   #4,PARSE_INPUT
                               0000V 31    02607          BRW     126$
       03 00000000G EF               00    E0 0260A 101$: BBS     #0,FULL_CHOICE,..+3              ; 5765
                               0000V 31    02612          BRW     107$
                    00000000   8F    DF 02615          PUSHAL  #0                                      ; 5769
          00000000G EF               01    FB 0261B    CALLS   #1,CLEAR
       00V00000000G EF               00    E0 02622          BBS     #0,FULL_PROMPT,104$              ; 5771
       03 00000000G EF               00    E0 0262A          BBS     #0,TEMP_FULL_PROMPT,..+3
                               0000V 31    02632          BRW     105$
                    00000000G  EF    9F 02635 104$: PUSHAB  SHIFT                                   ; 5775
                               04    DD 02638          PUSHL   #4
                    00000000G  EF    9F 0263D          PUSHAB  PAS$FV_OUTPUT
          00000000G EF               03    FB 02643    CALLS   #3,PAS$WRITE_STRING
                    FFFF6A14   EF    9F 0264A          PUSHAB  C.ATJ
                               02    DD 02650          PUSHL   #2
                    00000000G  EF    9F 02652          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 02658    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 0265F          PUSHAB  ANSI_REVERSE
                               04    DD 02665          PUSHL   #4
                    00000000G  EF    9F 02667          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 0266D    CALLS   #3,PAS$WRITE_STRING
                    FFFF69EE   EF    9F 02674          PUSHAB  C.ATK
                               0E    DD 0267A          PUSHL   #14
                    00000000G  EF    9F 0267C          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 02682    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02689          PUSHAB  SEC_ATTR
                               16    DD 0268F          PUSHL   #22
                    00000000G  EF    9F 02691          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 02697    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 0269E          PUSHAB  ANSI_RESET
                               04    DD 026A4          PUSHL   #4
                    00000000G  EF    9F 026A6          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 026AC    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 026B3          PUSHAB  CRLF
                               02    DD 026B9          PUSHL   #2
                    00000000G  EF    9F 026BB          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 026C1    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 026C8          PUSHAB  CRLF_SHIFT
                               06    DD 026CE          PUSHL   #6
                    00000000G  EF    9F 026D0          PUSHAB  PAS$FV_OUTPUT
 00000000G EF                  03    FB 026D6    CALLS   #3,PAS$WRITE_STRING
                    FFFF6995   EF    9F 026DD          PUSHAB  C.ATL
                               0F    DD 026E3          PUSHL   #15
```

```
                                Generated Code

                                00000000G  EF   9F 026E5          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 026EB          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 026F2          PUSHAB  CRLF_SHIFT
                                           06   DD 026F8          PUSHL   #6
                                00000000G  EF   9F 026FA          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 02700          CALLS   #3,PAS$WRITE_STRING
                                FFFF697B   EF   9F 02707          PUSHAB  C.ATM
                                           0C   DD 0270D          PUSHL   #12
                                00000000G  EF   9F 0270F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 02715          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 0271C          PUSHAB  CRLF_SHIFT
                                           06   DD 02722          PUSHL   #6
                                00000000G  EF   9F 02724          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0272A          CALLS   #3,PAS$WRITE_STRING
                                FFFF695D   EF   9F 02731          PUSHAB  C.ATN
                                           13   DD 02737          PUSHL   #19
                                00000000G  EF   9F 02739          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0273F          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 02746          PUSHAB  CRLF_SHIFT
                                           06   DD 0274C          PUSHL   #6
                                00000000G  EF   9F 0274E          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 02754          CALLS   #3,PAS$WRITE_STRING
                                FFFF6947   EF   9F 0275B          PUSHAB  C.ATO
                                           10   DD 02761          PUSHL   #16
                                00000000G  EF   9F 02763          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 02769          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 02770          PUSHAB  CRLF_SHIFT
                                           06   DD 02776          PUSHL   #6
                                00000000G  EF   9F 02778          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 0277E          CALLS   #3,PAS$WRITE_STRING
                                FFFF692D   EF   9F 02785          PUSHAB  C.ATP
                                           0C   DD 0278B          PUSHL   #12
                                00000000G  EF   9F 0278D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 02793          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 0279A          PUSHAB  CRLF_SHIFT
                                           06   DD 027A0          PUSHL   #6
                                00000000G  EF   9F 027A2          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 027A8          CALLS   #3,PAS$WRITE_STRING
                                FFFF690F   EF   9F 027AF          PUSHAB  C.ATQ
                                           0F   DD 027B5          PUSHL   #15
                                00000000G  EF   9F 027B7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 027BD          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 027C4          PUSHAB  CRLF_SHIFT
                                           06   DD 027CA          PUSHL   #6
                                00000000G  EF   9F 027CC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 027D2          CALLS   #3,PAS$WRITE_STRING
                                FFFF68F5   EF   9F 027D9          PUSHAB  C.ATR
                                           16   DD 027DF          PUSHL   #22
                                00000000G  EF   9F 027E1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 027E7          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 027EE          PUSHAB  CRLF
                                           02   DD 027F4          PUSHL   #2
                                00000000G  EF   9F 027F6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              03   FB 027FC          CALLS   #3,PAS$WRITE_STRING
                                00000000G  EF   9F 02803          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                              01   FB 02809          CALLS   #1,PAS$WRITELN2
                                      0000V    31 02810          BRW     112$
```

```
                    00000000G  EF   9F 02813 105$:   PUSHAB  SHIFT                                   ; 5798
                               04   DD 02819         PUSHL   #4
                    00000000G  EF   9F 0281B         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 02821    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 02828         PUSHAB  QUES_HINT
                               1F   DD 0282E         PUSHL   #31
                    00000000G  EF   9F 02830         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 02836    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 0283D         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              01   FB 02843    CALLS   #1,PAS$WRITELN2
                         0000V     31   0284A         BRW     112$
                    00000000   8F   DF 0284D 107$:    PUSHAL  #0                                      ; 5806
          00000000G EF              01   FB 02853    CALLS   #1,CLEAR
       00V000000000G EF              00   E0 0285A    BBS     #0,FULL_PROMPT,109$                     ; 5808
    03 000000000G EF                 00   E0 02862    BBS     #0,TEMP_FULL_PROMPT,.+3
                         0000V     31   0286A         BRW     110$
                    00000000G  EF   9F 0286D 109$:    PUSHAB  SHIFT                                   ; 5812
                               04   DD 02873         PUSHL   #4
                    00000000G  EF   9F 02875         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 0287B    CALLS   #3,PAS$WRITE_STRING
                    FFFF6862   EF   9F 02882         PUSHAB  C.ATS
                               02   DD 02888         PUSHL   #2
                    00000000G  EF   9F 0288A         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 02890    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 02897         PUSHAB  ANSI_REVERSE
                               04   DD 0289D         PUSHL   #4
                    00000000G  EF   9F 0289F         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 028A5    CALLS   #3,PAS$WRITE_STRING
                    FFFF683A   EF   9F 028AC         PUSHAB  C.ATT
                               10   DD 028B2         PUSHL   #16
                    00000000G  EF   9F 028B4         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 028BA    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 028C1         PUSHAB  SEC_ATTR
                               16   DD 028C7         PUSHL   #22
                    00000000G  EF   9F 028C9         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 028CF    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 028D6         PUSHAB  ANSI_RESET
                               04   DD 028DC         PUSHL   #4
                    00000000G  EF   9F 028DE         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 028E4    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 028EB         PUSHAB  CRLF
                               02   DD 028F1         PUSHL   #2
                    00000000G  EF   9F 028F3         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              03   FB 028F9    CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF   9F 02900         PUSHAB  PAS$FV_OUTPUT
          00000000G EF              01   FB 02906    CALLS   #1,PAS$WRITELN2
                    000000FC   8F   DD 0290D         PUSHL   #252                                    ; 5819
                               07   DD 02913         PUSHL   #7
                               04   DD 02915         PUSHL   #4
                    00000000G  EF   9F 02917         PUSHAB  SYS$OUTPUT_NAME
                               0B   DD 0291D         PUSHL   #11
                               01   DD 0291F         PUSHL   #1
                    00000000G  EF   9F 02921         PUSHAB  FDL_DEST
          00000000G EF              07   FB 02927    CALLS   #7,PAS$OPEN2
                    00000000G  EF   9F 0292E         PUSHAB  FDL_DEST                                ; 5821
          00000000G EF              01   FB 02934    CALLS   #1,PAS$REWRITE2
                    00000000G  EF   9F 0293B         PUSHAB  TEST                                    ; 5823
```

```
                Generated Code                                  

                          01    FB 02941           CALLS   #1,SHOW_PRIMARY_SECTION
                00000000G EF    9F 02948           PUSHAB  FDL_DEST                        ; 5825
00000000G EF              01    FB 0294E           CALLS   #1,PAS$CLOSE2
                          00V   11 02955           BRB     112$
                00000000G EF    9F 02957 110$:     PUSHAB  SHIFT                           ; 5831
                          04    DD 0295D           PUSHL   #4
                00000000G EF    9F 0295F           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 02965           CALLS   #3,PAS$WRITE_STRING
                00000000G EF    9F 0296C           PUSHAB  QUES_HINT
                          1F    DD 02972           PUSHL   #31
                00000000G EF    9F 02974           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 0297A           CALLS   #3,PAS$WRITE_STRING
                00000000G EF    9F 02981           PUSHAB  PAS$FV_OUTPUT
00000000G EF              01    FB 02987           CALLS   #1,PAS$WRITELN2
                00000000G EF    9F 0298E 112$:     PUSHAB  SHIFT                           ; 5835
                          04    DD 02994           PUSHL   #4
                00000000G EF    9F 02996           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 0299C           CALLS   #3,PAS$WRITE_STRING
                FFFF6753  EF    9F 029A3           PUSHAB  C_ATU
                          22    DD 029A9           PUSHL   #34
                00000000G EF    9F 029AB           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 029B1           CALLS   #3,PAS$WRITE_STRING
                00000000G EF    9F 029B8           PUSHAB  ANSI_REVERSE
                          04    DD 029BE           PUSHL   #4
                00000000G EF    9F 029C0           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 029C6           CALLS   #3,PAS$WRITE_STRING
                FFFF674D  EF    9F 029CD           PUSHAB  C_ATV
                          03    DD 029D3           PUSHL   #3
                00000000G EF    9F 029D5           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 029DB           CALLS   #3,PAS$WRITE_STRING
                00000000G EF    9F 029E2           PUSHAB  ANSI_RESET
                          04    DD 029E8           PUSHL   #4
                00000000G EF    9F 029EA           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 029F0           CALLS   #3,PAS$WRITE_STRING
                FFFF6727  EF    9F 029F7           PUSHAB  C_ATW
                          03    DD 029FD           PUSHL   #3
                00000000G EF    9F 029FF           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 02A05           CALLS   #3,PAS$WRITE_STRING
                00000000  8F    DF 02A0C           PUSHAL  #0                              ; 5837
                      00  8F    9F 02A12           PUSHAB  #0
      F4    AD 00000000G  EF    9E 02A15           MOVAB   EDF$AB_SHARING_TABLE_STA,-12(FP)
                      F4  AD    9F 02A1D           PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 02A20           MOVAB   EDF$AB_SHARING_TABLE_KEY,-16(FP)
                      F0  AD    9F 02A28           PUSHAB  -16(FP)
00000000G EF                04    FB 02A2B           CALLS   #4,PARSE_INPUT
                        0000V   31 02A32           BRW     126$
03 00000000G EF             00    E0 02A35 113$:     BBS     #0,FULL_CHOICE,.+3              ; 5850
                        0000V   31 02A3D           BRW     119$
                00000000  8F    DF 02A40           PUSHAL  #0                              ; 5854
00000000G EF                01    FB 02A46           CALLS   #1,CLEAR
00V00000000G EF             00    E0 02A4D           BBS     #0,FULL_PROMPT,116$             ; 5856
03 00000000G EF             00    E0 02A55           BBS     #0,TEMP_FULL_PROMPT, +3
                        0000V   31 02A5D           BRW     117$
                00000000G EF    9F 02A60 116$:     PUSHAB  SHIFT                           ; 5860
                          04    DD 02A66           PUSHL   #4
                00000000G EF    9F 02A68           PUSHAB  PAS$FV_OUTPUT
00000000G EF              03    FB 02A6E           CALLS   #3,PAS$WRITE_STRING
```

```
                 FFFF66AD   EF   9F 02A75          PUSHAB  C.ATX
                            02   DD 02A7B          PUSHL   #2
                 00000000G  EF   9F 02A7D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02A83          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02A8A          PUSHAB  ANSI_REVERSE
                            04   DD 02A90          PUSHL   #4
                 00000000G  EF   9F 02A92          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02A98          CALLS   #3,PAS$WRITE_STRING
                 FFFF6687   EF   9F 02A9F          PUSHAB  C.ATY
                            0D   DD 02AA5          PUSHL   #13
                 00000000G  EF   9F 02AA7          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02AAD          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02AB4          PUSHAB  SEC_ATTR
                            16   DD 02ABA          PUSHL   #22
                 00000000G  EF   9F 02ABC          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02AC2          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02AC9          PUSHAB  ANSI_RESET
                            04   DD 02ACF          PUSHL   #4
                 00000000G  EF   9F 02AD1          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02AD7          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02ADE          PUSHAB  CRLF
                            02   DD 02AE4          PUSHL   #2
                 00000000G  EF   9F 02AE6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02AEC          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02AF3          PUSHAB  CRLF_SHIFT
                            06   DD 02AF9          PUSHL   #6
                 00000000G  EF   9F 02AFB          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B01          CALLS   #3,PAS$WRITE_STRING
                 FFFF662E   EF   9F 02B08          PUSHAB  C.ATZ
                            0F   DD 02B0E          PUSHL   #15
                 00000000G  EF   9F 02B10          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B16          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02B1D          PUSHAB  CRLF_SHIFT
                            06   DD 02B23          PUSHL   #6
                 00000000G  EF   9F 02B25          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B2B          CALLS   #3,PAS$WRITE_STRING
                 FFFF6614   EF   9F 02B32          PUSHAB  C.AUA
                            10   DD 02B38          PUSHL   #16
                 00000000G  EF   9F 02B3A          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B40          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02B47          PUSHAB  CRLF_SHIFT
                            06   DD 02B4D          PUSHL   #6
                 00000000G  EF   9F 02B4F          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B55          CALLS   #3,PAS$WRITE_STRING
                 FFFF65FA   EF   9F 02B5C          PUSHAB  C.AUB
                            10   DD 02B62          PUSHL   #16
                 00000000G  EF   9F 02B64          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B6A          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02B71          PUSHAB  CRLF
                            02   DD 02B77          PUSHL   #2
                 00000000G  EF   9F 02B79          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 02B7F          CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF   9F 02B86          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               01   FB 02B8C          CALLS   #1,PAS$WRITELN2
                       0000V 31   02B93          BRW     124$
                 00000000G  EF   9F 02B96 117$:    PUSHAB  SHIFT                   ; 5875
                            04   DD 02B9C          PUSHL   #4
```

```
                    00000000G  EF    9F 02B9E          PUSHAB  PAS$FV_OUTPUT
                00C000000G  EF        03 FB 02BA4       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02BAB          PUSHAB  QUES_HINT
                               1F    DD 02BB1          PUSHL   #31
                    00000000G  EF    9F 02BB3          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03 FB 02BB9       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02BC0          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        01 FB 02BC6       CALLS   #1,PAS$WRITELN2
                              0000V  31 02BCD          BRW     124$
                    00000000   8F    DF 02BD0 119$:    PUSHAL  #0                                  ; 5883
                00000000G  EF        01 FB 02BD6       CALLS   #1,CLEAR
            00V00000000G  EF         00 E0 02BDD       BBS     #0,FULL_PROMPT,121$                 ; 5885
            03 00000000G  EF         00 E0 02BE5       BBS     #0,TEMP_FULL_PROMPT,.+3
                              0000V  31 02BED          BRW     122$
                    00000000G  EF    9F 02BF0 121$:    PUSHAB  SHIFT                               ; 5889
                               04    DD 02BF6          PUSHL   #4
                    00000000G  EF    9F 02BF8          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03 FB 02BFE       CALLS   #3,PAS$WRITE_STRING
                    FFFF6561   EF    9F 02C05          PUSHAB  C.AUC
                               02    DD 02C0B          PUSHL   #2
                    00000000G  EF    9F 02C0D          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03 FB 02C13       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02C1A          PUSHAB  ANSI_REVERSE
                               04    DD 02C20          PUSHL   #4
                    00000000G  EF    9F 02C22          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03 FB 02C28       CALLS   #3,PAS$WRITE_STRING
                    FFFF653B   EF    9F 02C2F          PUSHAB  C.AUD
                               0F    DD 02C35          PUSHL   #15
                    00000000G  EF    9F 02C37          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF        03 FB 02C3D       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02C44          PUSHAB  SEC_ATTR
                               16    DD 02C4A          PUSHL   #22
                    00000000G  EF    9F 02C4C          PUSHAB  PAS$FV_OUTPUT
            00000000G  EF            03 FB 02C52       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02C59          PUSHAB  ANSI_RESET
                               04    DD 02C5F          PUSHL   #4
                    00000000G  EF    9F 02C61          PUSHAB  PAS$FV_OUTPUT
            00000000G  EF            03 FB 02C67       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02C6E          PUSHAB  CRLF
                               02    DD 02C74          PUSHL   #2
                    00000000G  EF    9F 02C76          PUSHAB  PAS$FV_OUTPUT
            00000000G  EF            03 FB 02C7C       CALLS   #3,PAS$WRITE_STRING
                    00000000G  EF    9F 02C83          PUSHAB  PAS$FV_OUTPUT
            00000000G  EF            01 FB 02C89       CALLS   #1,PAS$WRITELN2
                    000000FC   8F    DD 02C90          PUSHL   #252                                ; 5896
                               07    DD 02C96          PUSHL   #7
                               04    DD 02C98          PUSHL   #4
                    00000000G  EF    9F 02C9A          PUSHAB  SYS$OUTPUT_NAME
                               0B    DD 02CA0          PUSHL   #11
                               01    DD 02CA2          PUSHL   #1
                    00000000G  EF    9F 02CA4          PUSHAB  FDL_DEST
            00000000G  EF            07 FB 02CAA       CALLS   #7,PAS$OPEN2
                    00000000G  EF    9F 02CB1          PUSHAB  FDL_DEST                            ; 5898
            00000000G  EF            01 FB 02CB7       CALLS   #1,PAS$REWRITE2
                    00000000G  EF    9F 02CBE          PUSHAB  TEST                                ; 5900
            00000000G  EF            01 FB 02CC4       CALLS   #1,SHOW_PRIMARY_SECTION
                    00000000G  EF    9F 02CCB          PUSHAB  FDL_DEST                            ; 5902
```

```
00000000G  EF                01  FB 02CD1          CALLS   #1,PAS$CLOSE2
                             00V 11 02CD8          BRB     124$
                 00000000G   EF  9F 02CDA 122$:     PUSHAB  SHIFT                          ; 5908
                             04  DD 02CE0          PUSHL   #4
                 00000000G   EF  9F 02CE2          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02CE8          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 02CEF          PUSHAB  QUES_HINT
                             1F  DD 02CF5          PUSHL   #31
                 00000000G   EF  9F 02CF7          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02CFD          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 02D04          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          01  FB 02D0A          CALLS   #1,PAS$WRITELN2
                 00000000G   EF  9F 02D11 124$:     PUSHAB  SHIFT                          ; 5912
                             04  DD 02D17          PUSHL   #4
                 00000000G   EF  9F 02D19          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D1F          CALLS   #3,PAS$WRITE_STRING
                 FFFF6454    EF  9F 02D26          PUSHAB  C.AUE
                             21  DD 02D2C          PUSHL   #33
                 00000000G   EF  9F 02D2E          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D34          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 02D3B          PUSHAB  ANSI_REVERSE
                             04  DD 02D41          PUSHL   #4
                 00000000G   EF  9F 02D43          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D49          CALLS   #3,PAS$WRITE_STRING
                 FFFF644E    EF  9F 02D50          PUSHAB  C.AUF
                             03  DD 02D56          PUSHL   #3
                 00000000G   EF  9F 02D58          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D5E          CALLS   #3,PAS$WRITE_STRING
                 00000000G   EF  9F 02D65          PUSHAB  ANSI_RESET
                             04  DD 02D6B          PUSHL   #4
                 00000000G   EF  9F 02D6D          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D73          CALLS   #3,PAS$WRITE_STRING
                 FFFF6428    EF  9F 02D7A          PUSHAB  C.AUG
                             03  DD 02D80          PUSHL   #3
                 00000000G   EF  9F 02D82          PUSHAB  PAS$FV_OUTPUT
00000000G        EF          03  FB 02D88          CALLS   #3,PAS$WRITE_STRING
                 00000000    8F  DF 02D8F          PUSHAL  #0                             ; 5914
                       00    8F  9F 02D95          PUSHAB  #0
      F4         AD 00000000G   EF  9E 02D98       MOVAB   EDF$AB_SYSTEM_TABLE_STA,-12(FP)
                       F4    AD  9F 02DA0          PUSHAB  -12(FP)
      F0         AD 00000000G   EF  9E 02DA3       MOVAB   EDF$AB_SYSTEM_TABLE_KEY,-16(FP)
                       F0    AD  9F 02DAB          PUSHAB  -16(FP)
00000000G        EF          04  FB 02DAE          CALLS   #4,PARSE_INPUT
                             00V 11 02DB5          BRB     126$
                                    02DB7 125$:
                 0F 00000019G   EF  91 02DB7 126$:  CMPB    TEST+25,#15                 ; 5929
                             00V 13 02DBE          BEQL    128$
00000000G        EF          01  90 02DC0          MOVB    #1,TEST                     ; 5931
0000001EG        EF 00000000G   EF  90 02DC7 128$:  MOVB    INPUT_VALUE,TEST+30         ; 5933
      87         8F 0000001EG   EF  91 02DD2       CMPB    TEST+30,#-121               ; 5940
                             00V 12 02DDA          BNEQ    130$
0000001FG        EF          07  D0 02DDC          MOVL    #7,TEST+31                  ; 5942
                             00V 11 02DE3          BRB     131$
0000001FG        EF 00000000G   EF  D0 02DE5 130$:  MOVL    EDF$GL_SECNUM,TEST+31       ; 5946
                    0000001FG   EF  D5 02DF0 131$:  TSTL    TEST+31                     ; 5948
                             00V 19 02DF6          BLSS    133$
                 07 0000001FG   EF  D1 02DF8       CMPL    TEST+31,#7
```

```
                                    00V  15 02DFF            BLEQ    134$
                                    00   DD 02E01 133$:      PUSHL   #0                                      ; 5954
                                    00   DD 02E03            PUSHL   #0
                                    00   DD 02E05            PUSHL   #0
                          00B38038  8F   DD 02E07            PUSHL   #11763768
    00000000G  EF                   04   FB 02E0D            CALLS   #4,LIB$SIGNAL
03  00000000G  EF                   00   E1 02E14 134$:      BBC     #0,FULL_CHOICE,..+3                     ; 5959
                                0000V  31 02E1C            BRW     147$
    00000000G  EF 00000000G         EF   D0 02E1F            MOVL    DEF_HEAD,DEF_CURRENT                    ; 5963
                        01          8F   9F 02E2A 136$:      PUSHAB  #1                                      ; 5967
                  00000000G         EF   9F 02E2D            PUSHAB  TEST
    00000000G  EF                   02   FB 02E33            CALLS   #2,CURRENT_EQ_TEST
               00V                  50   E8 02E3A            BLBS    R0,138$
    00000000G  EF                   00   FB 02E3D            CALLS   #0,INCR_CURRENT                         ; 5969
                           01       8F   9F 02E44 138$:      PUSHAB  #1
                  00000000G         EF   9F 02E47            PUSHAB  TEST
    00000000G  EF                   02   FB 02E4D            CALLS   #2,CURRENT_EQ_TEST
                                    51   94 02E54            CLRB    R1
                  5C 00000000G      EF   D0 02E56            MOVL    DEF_CURRENT,R12
                           01       AC   D5 02E5D            TSTL    1(R12)
                                    00V  12 02E60            BNEQ    140$
                                    51   96 02E62            INCB    R1
                  51                50   88 02E64 140$:      BISB2   R0,R1
                  C0                51   E9 02E67            BLBC    R1,136$
                     00000000G      EF   D5 02E6A            TSTL    DEF_CURRENT                             ; 5973
                                    00V  13 02E70            BEQL    145$
                           01       8F   9F 02E72            PUSHAB  #1                                      ; 5977
                     00000000G      EF   9F 02E75            PUSHAB  TEST
    00000000G  EF                   02   FB 02E7B            CALLS   #2,CURRENT_EQ_TEST
               00V                  50   E8 02E82            BLBS    R0,147$
                                    00   DD 02E85            PUSHL   #0                                      ; 5979
                                    00   DD 02E87            PUSHL   #0
                                    00   DD 02E89            PUSHL   #0
                          00B38038  8F   DD 02E8B            PUSHL   #11763768
    00000000G  EF                   04   FB 02E91            CALLS   #4,LIB$SIGNAL
                                    00V  11 02E98            BRB     147$
                                    00   DD 02E9A 145$:      PUSHL   #0                                      ; 5985
                                    00   DD 02E9C            PUSHL   #0
                                    00   DD 02E9E            PUSHL   #0
                          00B38038  8F   DD 02EA0            PUSHL   #11763768
    00000000G  EF                   04   FB 02EA6            CALLS   #4,LIB$SIGNAL
                     00000000G      EF   94 02EAD 147$:      CLRB    TEMP_FULL_PROMPT                        ; 5992
                                         04 02EB3            RET                                             ; 5994

; Routine Size: 11956 bytes,    Routine Base: $CODE + 0A990

                                         00000 ASK_TEST_SECONDARY_VALUE:                                      ; 6056
                                    0000 00000            .WORD   ^M<>
                  5E      FEED      CE   9E 00002            MOVAB   -275(SP),SP
                  0F 000000019G     EF   91 00007            CMPB    TEST+25,#15                             ; 6750
                                    00V  12 0000E            BNEQ    2$
                     00000000G      EF   94 00010            CLRB    TEST                                    ; 6752
                                    00V  11 00016            BRB     4$
    00000000G  EF                   01   90 00018 2$:        MOVB    #1,TEST                                 ; 6756
               51                   5D   D0 0001F 4$:        MOVL    FP,R1                                   ; 6763
            00V  AF                 00   FB 00022            CALLS   #0,THE_QUESTION
 F1 00000000G  EF                   00   E0 00026            BBS     #0,SYS$INPUT_ERROR,4$
```

```
                                 00000000G  EF   9F 0002E          PUSHAB  INPUT_DESC                                  ; 6767
                      00000000G  EF         01   FB 00034          CALLS   #1,STR$FREE1_DX
                                            04   00038          RET                                                 ; 6769

; Routine Size: 60 bytes,    Routine Base: $CODE + 0D844

                                            00000 THE_QUESTION:                                                      ; 6065
                                       001C 00000          .WORD   ^M<R2,R3,R4>
                               5E           10   C2 00002          SUBL2   #16,SP
                                      F8    AD   D4 00005          CLRL    -8(FP)
                               6D 00000000G  EF   9E 00008          MOVAB   PAS$HANDLER,(FP)
                               5C            51   D0 0000F          MOVL    R1,R12
                                  00000000G  EF   94 00012          CLRB    SYS$INPUT_ERROR                         ; 6072
                             F8   AD 00000000G  EF   9E 00018          MOVAB   SYS$INPUT_COND_HANDLER,FP-8             ; 6073
                   00V00000000G   EF         00   E1 00020          BBC     #0,TEMP_FULL_PROMPT,2$                  ; 6075
                                  00000000G  EF   9F 00028          PUSHAB  SHIFT                                   ; 6077
                                             04   DD 0002E          PUSHL   #4
                                  00000000G  EF   9F 00030          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 00036          CALLS   #3,PAS$WRITE_STRING
                                  FFFF6279   EF   9F 0003D          PUSHAB  C.AUH
                                             32   DD 00043          PUSHL   #50
                                  00000000G  EF   9F 00045          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 0004B          CALLS   #3,PAS$WRITE_STRING
                                  00000000G  EF   9F 00052          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         01   FB 00058          CALLS   #1,PAS$WRITELN2
                               07 00000019G  EF   91 0005F 2$:       CMPB    TEST+25,#7                              ; 6083
                                        00V  12   00066          BNEQ    4$
                                  00000000G  EF   9F 00068          PUSHAB  CRLF_SHIFT                              ; 6085
                                             06   DD 0006E          PUSHL   #6
                                  00000000G  EF   9F 00070          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 00076          CALLS   #3,PAS$WRITE_STRING
                                  FFFF626D   EF   9F 0007D          PUSHAB  C.AUI
                                             19   DD 00083          PUSHL   #25
                                  00000000G  EF   9F 00085          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 0008B          CALLS   #3,PAS$WRITE_STRING
                               21 0000001EG  EF   91 00092 4$:       CMPB    TEST+30,#33                             ; 6088
                                        03   13   00099          BEQL    .+3
                                       0000V 31   0009B          BRW     6$
                                  00000000G  EF   9F 0009E          PUSHAB  CRLF_SHIFT                              ; 6090
                                             06   DD 000A4          PUSHL   #6
                                  00000000G  EF   9F 000A6          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 000AC          CALLS   #3,PAS$WRITE_STRING
                                  FFFF6253   EF   9F 000B3          PUSHAB  C.AUJ
                                             28   DD 000B9          PUSHL   #40
                                  00000000G  EF   9F 000BB          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 000C1          CALLS   #3,PAS$WRITE_STRING
                                  00000000G  EF   9F 000C8          PUSHAB  CRLF_SHIFT
                                             06   DD 000CE          PUSHL   #6
                                  00000000G  EF   9F 000D0          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 000D6          CALLS   #3,PAS$WRITE_STRING
                                  FFFF6251   EF   9F 000DD          PUSHAB  C.AUK
                                             16   DD 000E3          PUSHL   #22
                                  00000000G  EF   9F 000E5          PUSHAB  PAS$FV_OUTPUT
                      00000000G   EF         03   FB 000EB          CALLS   #3,PAS$WRITE_STRING
                                  00000000G  EF   9F 000F2          PUSHAB  CRLF_SHIFT
                                             06   DD 000F8          PUSHL   #6
                                  00000000G  EF   9F 000FA          PUSHAB  PAS$FV_OUTPUT
```

```
                00000000G  EF                    03  FB 00100          CALLS   #3,PAS$WRITE_STRING
                               FFFF623F          EF  9F 00107          PUSHAB  C.AUL
                                                 1A  DD 0010D          PUSHL   #26
                               00000000G         EF  9F 0010F          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                    03  FB 00115          CALLS   #3,PAS$WRITE_STRING
                                                00V  11 0011C          BRB     9$
                           50 0000001EG         EF  9A 0011E 6$:       MOVZBL  TEST+30,R0                      ; 6095
                           50                   04  C4 00125          MULL2   #4,R0
                           50                   02  C0 00128          ADDL2   #2,R0
           00V00000000G    EF                   50  E0 0012B          BBS     R0,SEC_TYPE,8$
                               00000000G        EF  9F 00133          PUSHAB  CRLF_SHIFT                      ; 6097
                                                06  DD 00139          PUSHL   #6
                               00000000G        EF  9F 0013B          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                    03  FB 0C141          CALLS   #3,PAS$WRITE_STRING
                               FFFF621A         EF  9F 00148          PUSHAB  C.AUM
                                                20  DD 0014E          PUSHL   #32
                               00000000G        EF  9F 00150          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                    03  FB 00156          CALLS   #3,PAS$WRITE_STRING
                                                    00015D 8$:
                           50 0000001EG         EF  9A 0015D 9$:       MOVZBL  TEST+30,R0                      ; 6099
                           50                   04  C4 00164          MULL2   #4,R0
                           50                   02  C0 00167          ADDL2   #2,R0
           03 00000000G    EF                   50  E0 0016A          BBS     R0,SEC_TYPE,.+3
                                              0000V 31 00172          BRW     27$
                           50 0000001EG         EF  9A 00175          MOVZBL  TEST+30,R0                      ; 6103
           34      62      8F                   50  8F 0017C          CASEB   R0,#98,#52
                                             0000V     00181          .DISPL  11$
                                             006A      00183          .DISPL  106
                                             006A      00185          .DISPL  106
                                             006A      00187          .DISPL  106
                                             006A      00189          .DISPL  106
                                             006A      0018B          .DISPL  106
                                             006A      0018D          .DISPL  106
                                             006A      0018F          .DISPL  106
                                             006A      00191          .DISPL  106
                                             006A      00193          .DISPL  106
                                             006A      00195          .DISPL  106
                                             006A      00197          .DISPL  106
                                             006A      00199          .DISPL  106
                                             006A      0019B          .DISPL  106
                                             006A      0019D          .DISPL  106
                                             006A      0019F          .DISPL  106
                                             006A      001A1          .DISPL  106
                                             006A      001A3          .DISPL  106
                                             006A      001A5          .DISPL  106
                                             006A      001A7          .DISPL  106
                                             0000V     001A9          .DISPL  13$
                                             006A      001AB          .DISPL  106
                                             006A      001AD          .DISPL  106
                                             006A      001AF          .DISPL  106
                                             006A      001B1          .DISPL  106
                                             006A      001B3          .DISPL  106
                                             006A      001B5          .DISPL  106
                                             006A      001B7          .DISPL  106
                                             006A      001B9          .DISPL  106
                                             006A      001BB          .DISPL  106
                                             006A      001BD          .DISPL  106
```

```
                                 006A      001BF          .DISPL  106
                                 006A      001C1          .DISPL  106
                                 006A      001C3          .DISPL  106
                                 006A      001C5          .DISPL  106
                                 006A      001C7          .DISPL  106
                                 006A      001C9          .DISPL  106
                                 0000V     001CB          .DISPL  16$
                                 006A      001CD          .DISPL  106
                                 0000V     001CF          .DISPL  14$
                                 006A      001D1          .DISPL  106
                                 0000V     001D3          .DISPL  15$
                                 006A      001D5          .DISPL  106
                                 006A      001D7          .DISPL  106
                                 006A      001D9          .DISPL  106
                                 006A      001DB          .DISPL  106
                                 006A      001DD          .DISPL  106
                                 006A      001DF          .DISPL  106
                                 006A      001E1          .DISPL  106
                                 006A      001E3          .DISPL  106
                                 006A      001E5          .DISPL  106
                                 0000V     001E7          .DISPL  12$
                                 0000V     001E9          .DISPL  13$
                                 0000V  31 001EB          BRW     17$
                    00000000G    EF   9F 001EE 11$:      PUSHAB  CRLF_SHIFT                      ; 6107
                                 06   DD 001F4          PUSHL   #6
                    00000000G    EF   9F 001F6          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 001FC          CALLS   #3,PAS$WRITE_STRING
                    FFFF617F     EF   9F 00203          PUSHAB  C.AUN
                                 1D   DD 00209          PUSHL   #29
                    00000000G    EF   9F 0020B          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00211          CALLS   #3,PAS$WRITE_STRING
                              0000V  31 00218          BRW     18$
                    00000000G    EF   9F 0021B 12$:      PUSHAB  CRLF_SHIFT                      ; 6113
                                 06   DD 00221          PUSHL   #6
                    00000000G    EF   9F 00223          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00229          CALLS   #3,PAS$WRITE_STRING
                    FFFF6172     EF   9F 00230          PUSHAB  C.AUO
                                 2F   DD 00236          PUSHL   #47
                    00000000G    EF   9F 00238          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0023E          CALLS   #3,PAS$WRITE_STRING
                              0000V  31 00245          BRW     18$
                    00000000G    EF   9F 00248 13$:      PUSHAB  CRLF_SHIFT                      ; 6118
                                 06   DD 0024E          PUSHL   #6
                    00000000G    EF   9F 00250          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00256          CALLS   #3,PAS$WRITE_STRING
                    FFFF6175     EF   9F 0025D          PUSHAB  C.AUP
                                 3E   DD 00263          PUSHL   #62
                    00000000G    EF   9F 00265          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 0026B          CALLS   #3,PAS$WRITE_STRING
                              0000V  31 00272          BRW     18$
                    00000000G    EF   9F 00275 14$:      PUSHAB  CRLF_SHIFT                      ; 6123
                                 06   DD 0027B          PUSHL   #6
                    00000000G    EF   9F 0027D          PUSHAB  PAS$FV_OUTPUT
00000000G  EF                    03   FB 00283          CALLS   #3,PAS$WRITE_STRING
                    FFFF6188     EF   9F 0028A          PUSHAB  C.AUQ
                                 24   DD 00290          PUSHL   #36
                    00000000G    EF   9F 00292          PUSHAB  PAS$FV_OUTPUT
```

```
                                                
```

```
                    006A      003B2          .DISPL  106
                    006A      003B4          .DISPL  106
                    006A      003B6          .DISPL  106
                    006A      003B8          .DISPL  106
                    006A      003BA          .DISPL  106
                    006A      003BC          .DISPL  106
                    006A      003BE          .DISPL  106
                    006A      003C0          .DISPL  106
                    006A      003C2          .DISPL  106
                    006A      003C4          .DISPL  106
                    006A      003C6          .DISPL  106
                    006A      003C8          .DISPL  106
                    006A      003CA          .DISPL  106
                    006A      003CC          .DISPL  106
                    006A      003CE          .DISPL  106
                    006A      003D0          .DISPL  106
                    006A      003D2          .DISPL  106
                    006A      003D4          .DISPL  106
                    0000V     003D6          .DISPL  21$
                    006A      003D8          .DISPL  106
                    006A      003DA          .DISPL  106
                    006A      003DC          .DISPL  106
                    006A      003DE          .DISPL  106
                    006A      003E0          .DISPL  106
                    006A      003E2          .DISPL  106
                    006A      003E4          .DISPL  106
                    006A      003E6          .DISPL  106
                    006A      003E8          .DISPL  106
                    006A      003EA          .DISPL  106
                    006A      003EC          .DISPL  106
                    006A      003EE          .DISPL  106
                    006A      003F0          .DISPL  106
                    006A      003F2          .DISPL  106
                    006A      003F4          .DISPL  106
                    006A      003F6          .DISPL  106
                    0000V     003F8          .DISPL  24$
                    006A      003FA          .DISPL  106
                    0000V     003FC          .DISPL  22$
                    006A      003FE          .DISPL  106
                    0000V     00400          .DISPL  23$
                    006A      00402          .DISPL  106
                    006A      00404          .DISPL  106
                    006A      00406          .DISPL  106
                    006A      00408          .DISPL  106
                    006A      0040A          .DISPL  106
                    006A      0040C          .DISPL  106
                    006A      0040E          .DISPL  106
                    006A      00410          .DISPL  106
                    006A      00412          .DISPL  106
                    0000V     00414          .DISPL  20$
                    0000V     00416          .DISPL  20$
                    0000V  31 00418          BRW     25$
          00000000    8F    DF 0041B 19$:    PUSHAL  #0                              ; 6150
                00    8F    9F 00421         PUSHAB  #0
F4  AD 00000000G    EF    9E 00424         MOVAB   EDF$AB_ORG_TABLE_STA,-12(FP)
                F4  AD    9F 0042C         PUSHAB  -12(FP)
F0  AD 00000000G    EF    9E 0042F         MOVAB   EDF$AB_ORG_TABLE_KEY,-16(FP)
```

```
                    F0    AD    9F 00437          PUSHAB  -16(FP)
00000000G  EF             04    FB 0043A          CALLS   #4,PARSE_INPUT
                    0000V 31    00441             BRW     26$
                00000000  8F    DF 00444 20$:     PUSHAL  #0                                      ; 6160
                      00  8F    9F 0044A          PUSHAB  #0
      F4    AD 00000000G  EF    9E 0044D          MOVAB   EDF$AB_SOURCE_TABLE_STA,-12(FP)
                      F4  AD    9F 00455          PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 00458          MOVAB   EDF$AB_SOURCE_TABLE_KEY,-16(FP)
                      F0  AD    9F 00460          PUSHAB  -16(FP)
00000000G  EF             04    FB 00463          CALLS   #4,PARSE_INPUT
                    0000V 31    0046A             BRW     26$
                00000000  8F    DF 0046D 21$:     PUSHAL  #0                                      ; 6169
                      00  8F    9F 00473          PUSHAB  #0
      F4    AD 00000000G  EF    9E 00476          MOVAB   EDF$AB_RU_TABLE_STA,-12(FP)
                      F4  AD    9F 0047E          PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 00481          MOVAB   EDF$AB_RU_TABLE_KEY,-16(FP)
                      F0  AD    9F 00489          PUSHAB  -16(FP)
00000000G  EF             04    FB 0048C          CALLS   #4,PARSE_INPUT
                    00V   11    00493             BRB     26$
                00000000  8F    DF 00495 22$:     PUSHAL  #0                                      ; 6178
                      00  8F    9F 0049B          PUSHAB  #0
      F4    AD 00000000G  EF    9E 0049E          MOVAB   EDF$AB_CARR_TABLE_STA,-12(FP)
                      F4  AD    9F 004A6          PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 004A9          MOVAB   EDF$AB_CARR_TABLE_KEY,-16(FP)
                      F0  AD    9F 004B1          PUSHAB  -16(FP)
00000000G  EF             04    FB 004B4          CALLS   #4,PARSE_INPUT
                    00V   11    004BB             BRB     26$
                00000000  8F    DF 004BD 23$:     PUSHAL  #0                                      ; 6187
                      00  8F    9F 004C3          PUSHAB  #0
      F4    AD 00000000G  EF    9E 004C6          MOVAB   EDF$AB_FORMAT_TABLE_STA,-12(FP)
                      F4  AD    9F 004CE          PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 004D1          MOVAB   EDF$AB_FORMAT_TABLE_KEY,-16(FP)
                      F0  AD    9F 004D9          PUSHAB  -16(FP)
   00000000G  EF             04    FB 004DC          CALLS   #4,PARSE_INPUT
                    00V   11    004E3             BRB     26$
                00000000  8F    DF 004E5 24$:     PUSHAL  #0                                      ; 6196
                      00  8F    9F 004EB          PUSHAB  #0
      F4    AD 00000000G  EF    9E 004EE          MOVAB   EDF$AB_TYPE_TABLE_STA,-12(FP)
                      F4  AD    9F 004F6          PUSHAB  -12(FP)
      F0    AD 00000000G  EF    9E 004F9          MOVAB   EDF$AB_TYPE_TABLE_KEY,-16(FP)
                      F0  AD    9F 00501          PUSHAB  -16(FP)
   00000000G  EF             04    FB 00504          CALLS   #4,PARSE_INPUT
                    00V   11    0050B             BRB     26$
                                0050D 25$:
   00000023G  EF 00000000G  EF    D0 0050D 26$:     MOVL    INPUT_VALUE,TEST+35                     ; 6209
                 50 0000001EG  EF    9A 00518 27$:     MOVZBL  TEST+30,R0                      ; 6213
                 50            04    C4 0051F          MULL2   #4,R0
                               50    D6 00522          INCL    R0
03 00000000G  EF              50    E0 00524          BBS     R0,SEC_TYPE,.+3
                        0000V 31    0052C             BRW     65$
                 50 0000001EG  EF    9A 0052F          MOVZBL  TEST+30,R0                      ; 6217
         EC     AC 00000000GEF40    D0 00536          MOVL    SECONDARY_MAX[R0],-20(R12)
                 50 0000001EG  EF    9A 0053F          MOVZBL  TEST+30,R0                      ; 6219
   00000098     8F            50    D1 00546          CMPL    R0,#152
                              00V   1E 0054D          BGEQU   30$
00VFFFF5F81     EF            50    E1 0054F          BBC     R0,C.AUX,30$
                    FFFF5F8F  EF    9F 00557          PUSHAB  C.AUY                                   ; 6227
```

```
                                    08   DD 0055D          PUSHL   #8
                        00000000G   EF   9F 0055F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 00565          CALLS   #3,PAS$WRITE_STRING
                                    00V  11 0056C          BRB     34$
3B9AC9FF    8F          EC          AC   D1 0056E 30$:     CMPL    -20(R12),#999999999                ; 6229
                                    00V  12 00576          BNEQ    32$
                        FFFF5F76    EF   9F 00578          PUSHAB  C.AUZ                              ; 6231
                                    08   DD 0057E          PUSHL   #8
                        00000000G   EF   9F 00580          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 00586          CALLS   #3,PAS$WRITE_STRING
                                    00V  11 0058D          BRB     34$
                        FFFF5F67    EF   9F 0058F 32$:     PUSHAB  C.AVA                              ; 6235
                                    02   DD 00595          PUSHL   #2
                        00000000G   EF   9F 00597          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 0059D          CALLS   #3,PAS$WRITE_STRING
                              EC    AC   9F 005A4          PUSHAB  -20(R12)
00000000G   EF                      01   FB 005A7          CALLS   #1,NUM_LEN
                                    50   DD 005AE          PUSHL   R0
                              EC    AC   DD 005B0          PUSHL   -20(R12)
                        00000000G   EF   9F 005B3          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 005B9          CALLS   #3,PAS$WRITE_INTEGER
                                    01   DD 005C0          PUSHL   #1
                                    29   DD 005C2          PUSHL   #41
                        00000000G   EF   9F 005C4          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 005CA          CALLS   #3,PAS$WRITE_CHAR
                        00000000G   EF   9F 005D1 34$:     PUSHAB  ANSI_REVERSE                       ; 6237
                                    04   DD 005D7          PUSHL   #4
                        00000000G   EF   9F 005D9          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 005DF          CALLS   #3,PAS$WRITE_STRING
                        FFFF5F14    EF   9F 005E6          PUSHAB  C.AVB
                                    03   DD 005EC          PUSHL   #3
                        00000000G   EF   9F 005EE          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 005F4          CALLS   #3,PAS$WRITE_STRING
                        00000000G   EF   9F 005FB          PUSHAB  ANSI_RESET
                                    04   DD 00601          PUSHL   #4
                        00000000G   EF   9F 00603          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 00609          CALLS   #3,PAS$WRITE_STRING
                                    52   94 00610          CLRB    R2                                 ; 6239
                              EC    AC   9F 00612          PUSHAB  -20(R12)
00000000G   EF                      01   FB 00615          CALLS   #1,NUM_LEN
                                    50   D1 0061C          CMPL    R0,#8
                                    00V  15 0061F          BLEQ    36$
                                    52   96 00621          INCB    R2
                                    50   94 00623 36$:     CLRB    R0
3B9AC9FF    8F          EC          AC   D1 00625          CMPL    -20(R12),#999999999
                                    00V  12 0062D          BNEQ    38$
                                    50   96 0062F          INCB    R0
                        52                  50   8A 00631 38$:     BICB2   R0,R2
                        00V                 52   E9 00634          BLBC    R2,40$
                        FFFF5EC7    EF   9F 00637          PUSHAB  C.AVC                              ; 6245
                                    03   DD 0063D          PUSHL   #3
                        00000000G   EF   9F 0063F          PUSHAB  PAS$FV_OUTPUT
00000000G   EF                      03   FB 00645          CALLS   #3,PAS$WRITE_STRING
                                    00V  11 0064C          BRB     41$
                        FFFF5EB4    EF   9F 0064E 40$:     PUSHAB  C.AVD                              ; 6249
                                    03   DD 00654          PUSHL   #3
                        00000000G   EF   9F 00656          PUSHAB  PAS$FV_OUTPUT
```

```
                00000000G  EF                 03   FB 0065C              CALLS   #3,PAS$WRITE_STRING
                           00000000   8F       DF   00663 41$:           PUSHAL  #0                          ; 6251
                                 00   8F       9F   00669                PUSHAB  #0
                           00000027G  EF       9F   0066C                PUSHAB  TEST+39
                00000000G  EF                 03   FB 00672              CALLS   #3,NUMBER_INPUT
                           50 0000001EG  EF     9A   00679              MOVZBL  TEST+30,R0                  ; 6253
                00000098   8F                 50   D1 00680              CMPL    R0,#152
                                              00V  1E 00687              BGEQU   50$
            00VFFFF5E7B    EF                 50   E1 00689              BBC     R0,C.AVE,50$
            00V00000033G   EF                 00   E1 00691              BBC     #0,VDATA+51,46$             ; 6263
                           03 000000F8G  EF     D1   00699              CMPL    IDATA+248,#3
                                              00V  18 006A0              BGEQ    46$
                           00000027G  EF       D5   006A2              TSTL    TEST+39
                                              00V  13 006A8              BEQL    46$
                                              00   DD 006AA              PUSHL   #0                          ; 6270
                                              00   DD 006AC              PUSHL   #0
                                              00   DD 006AE              PUSHL   #0
                           00B38038   8F       DD   006B0              PUSHL   #11763768
                00000000G  EF                 04   FB 006B6              CALLS   #4,LIB$SIGNAL
                      EC   AC 00000027G  EF     D1   006BD 46$:           CMPL    TEST+39,-20(R12)            ; 6272
                                              00V  14 006C5              BGTR    48$
                           50         EC   AC   CE   006C7              MNEGL   -20(R12),R0
                           50 00000027G  EF     D1   006CB              CMPL    TEST+39,R0
                                              00V  18 006D2              BGEQ    60$
                                              00   DD 006D4 48$:           PUSHL   #0                          ; 6278
                                              00   DD 006D6              PUSHL   #0
                                              00   DD 006D8              PUSHL   #0
                           00B38038   8F       DD   006DA              PUSHL   #11763768
                00000000G  EF                 04   FB 006E0              CALLS   #4,LIB$SIGNAL
                                              00V  11 006E7              BRB     60$
                      8A   8F 0000001EG  EF     91   006E9 50$:           CMPB    TEST+30,#-118               ; 6282
                                              00V  12 006F1              BNEQ    55$
                           01 00000027G  EF     D1   006F3              CMPL    TEST+39,#1                  ; 6286
                                              00V  19 006FA              BLSS    53$
                      EC   AC 00000027G  EF     D1   006FC              CMPL    TEST+39,-20(R12)
                                              00V  15 00704              BLEQ    59$
                                              00   DD 00706 53$:           PUSHL   #0                          ; 6292
                                              00   DD 00708              PUSHL   #0
                                              00   DD 0070A              PUSHL   #0
                           00B38038   8F       DD   0070C              PUSHL   #11763768
                00000000G  EF                 04   FB 00712              CALLS   #4,LIB$SIGNAL
                                              00V  11 00719              BRB     59$
                           00000027G  EF       D5   0071B 55$:           TSTL    TEST+39                     ; 6300
                                              00V  19 00721              BLSS    57$
                      EC   AC 00000027G  EF     D1   00723              CMPL    TEST+39,-20(R12)
                                              00V  15 0072B              BLEQ    58$
                                              00   DD 0072D 57$:           PUSHL   #0                          ; 6306
                                              00   DD 0072F              PUSHL   #0
                                              00   DD 00731              PUSHL   #0
                           00B38038   8F       DD   00733              PUSHL   #11763768
                00000000G  EF                 04   FB 00739              CALLS   #4,LIB$SIGNAL
                                                   00740 58$:
                                                   00740 59$:
                      56   8F 0000001EG  EF     91   00740 60$:           CMPB    TEST+30,#86                 ; 6310
                                              00V  12 00748              BNEQ    65$
                           00000027G  EF       D5   0074A              TSTL    TEST+39
                                              00V  15 00750              BLEQ    65$
```

```
                    14 00000027G   EF   D1 00752          CMPL    TEST+39,#20
                               00V 18 00759          BGEQ    65$
                               00  DD 0075B          PUSHL   #0                                      ; 6318
                               00  DD 0075D          PUSHL   #0
                               00  DD 0075F          PUSHL   #0
                        00B38038   8F   DD 00761          PUSHL   #11763768
              00000000G EF         04   FB 00767          CALLS   #4,LIB$SIGNAL
                    50 0000001EG   EF   9A 0076E 65$:     MOVZBL  TEST+30,R0                          ; 6322
                    50             04   C4 00775          MULL2   #4,R0
           00V00000000G EF         50   E0 00778          BBS     R0,SEC_TYPE,67$
                    0F 00000019G   EF   91 00780          CMPB    TEST+25,#15
                               03   13 00787          BEQL    .+3
                                  0000V 31 00789          BRW     95$
                    07 00000019G   EF   91 0078C 67$:     CMPB    TEST+25,#7                          ; 6330
                               00V 12 00793          BNEQ    69$
                        FFFF5D85   EF   9F 00795          PUSHAB  C.AVF                               ; 6332
                               09   DD 00797          PUSHL   #9
                        00000000G  EF   9F 00799          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 007A3          CALLS   #3,PAS$WRITE_STRING
                        00000000G  EF   9F 007AA          PUSHAB  ANSI_REVERSE
                               04   DD 007B0          PUSHL   #4
                        00000000G  EF   9F 007B2          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 007B8          CALLS   #3,PAS$WRITE_STRING
                        FFFF5D67   EF   9F 007BF          PUSHAB  C.AVG
                               03   DD 007C5          PUSHL   #3
                        00000000G  EF   9F 007C7          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 007CD          CALLS   #3,PAS$WRITE_STRING
                        00000000G  EF   9F 007D4          PUSHAB  ANSI_RESET
                               04   DD 007DA          PUSHL   #4
                        00000000G  EF   9F 007DC          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 007E2          CALLS   #3,PAS$WRITE_STRING
                        FFFF5D41   EF   9F 007E9          PUSHAB  C.AVH
                               03   DD 007EF          PUSHL   #3
                        00000000G  EF   9F 007F1          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 007F7          CALLS   #3,PAS$WRITE_STRING
                               00V 11 007FE          BRB     73$
                    81 8F 0000001EG   EF   91 00800 69$:     CMPB    TEST+30,#-127                   ; 6334
                               00V 12 00808          BNEQ    71$
                        FFFF5D24   EF   9F 0080A          PUSHAB  C.AVI                               ; 6336
                               14   DD 00810          PUSHL   #20
                        00000000G  EF   9F 00812          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 00818          CALLS   #3,PAS$WRITE_STRING
                               00V 11 0081F          BRB     73$
                        FFFF5D21   EF   9F 00821 71$:     PUSHAB  C.AVJ                               ; 6340
                               12   DD 00827          PUSHL   #18
                        00000000G  EF   9F 00829          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 0082F          CALLS   #3,PAS$WRITE_STRING
                        00000000G  EF   9F 00836          PUSHAB  CRLF_SHIFT
                               06   DD 0083C          PUSHL   #6
                        00000000G  EF   9F 0083E          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 00844          CALLS   #3,PAS$WRITE_STRING
                        FFFF5D09   EF   9F 0084B          PUSHAB  C.AVK
                               02   DD 00851          PUSHL   #2
                        00000000G  EF   9F 00853          PUSHAB  PAS$FV_OUTPUT
              00000000G EF         03   FB 00859          CALLS   #3,PAS$WRITE_STRING
           00V00000000G EF         30   E0 00860 73$:     BBS     #48,PAS$FV_INPUT,74$                ; 6342
                        00000000G  EF   9F 00868          PUSHAB  PAS$FV_INPUT
```

```
                                    01 FB 0086E            CALLS   #1,PAS$LOOK_AHEAD
                                    31 E0 00875 74$:       BBS     #49,PAS$FV_INPUT,76$
                 00000000G  EF      9F 0087D               PUSHAB  PAS$FV_INPUT                        ; 6346
                                    01 FB 00883            CALLS   #1,PAS$RESET2
                                    00 DD 0088A            PUSHL   #0                                  ; 6347
                                    00 DD 0088C            PUSHL   #0
                                    00 DD 0088E            PUSHL   #0
                 00B3804B   8F      DD 00890               PUSHL   #11763787
                                    04 FB 00896            CALLS   #4,LIB$SIGNAL
                 000000FF   8F      DD 0089D 76$:          PUSHL   #255                                ; 6351
                 00000000G  EF      9F 008A3               PUSHAB  PAS$FV_INPUT
                     FEED   CC      9F 008A9               PUSHAB  -275(R12)
                                    03 FB 008AD            CALLS   #3,PAS$READ_STRING
                 00000000G  EF      9F 008B4               PUSHAB  PAS$FV_INPUT
                                    01 FB 008BA            CALLS   #1,PAS$READLN2
                 00000000G  EF      9F 008C1               PUSHAB  CRLF                                ; 6352
                                    02 DD 008C7            PUSHL   #2
                 00000000G  EF      9F 008C9               PUSHAB  PAS$FV_OUTPUT
                                    03 FB 008CF            CALLS   #3,PAS$WRITE_STRING
                 00000000G  EF      9F 008D6               PUSHAB  PAS$FV_OUTPUT
                                    01 FB 008DC            CALLS   #1,PAS$WRITELN2
                 00000000G  EF      7D 008E3               MOVQ    NULL_STRING,TEST+17                 ; 6354
                       F0   AD      DO 008EE               MOVL    #17694975,-16(FP)                   ; 6355
                       F4   AD  FEED   CC 9E 008F6         MOVAB   -275(R12),-12(FP)
                                F0   AD 9F 008FC           PUSHAB  -16(FP)
                 00000011G  EF      9F 008FF               PUSHAB  TEST+17
                                    02 FB 00905            CALLS   #2,STR$TRIM
                 00000000G  EF      9F 0090C               PUSHAB  INPUT_DESC                          ; 6356
                 00000011G  EF      9F 00912               PUSHAB  TEST+17
                                    02 FB 00918            CALLS   #2,LIB$SCOPY_DXDX
                 00000000G  EF  00000004G  EF  DO 0091F     MOVL    INPUT_DESC+4,PARAM_BLOCK+20         ; 6357
                 00000010G  EF  00000000G  EF  3C 0092A     MOVZWL  INPUT_DESC,PARAM_BLOCK+16           ; 6358
                                    00 E1 00935            BBC     #0,JOURNAL_ENABLED,82$              ; 6364
                 00000011G  EF      B5 0093D               TSTW    TEST+17                             ; 6366
                                    00V 1B 00943           BLEQU   80$
                 7E 00000011G  EF   3C 00945               MOVZWL  TEST+17,-(SP)                       ; 6368
                                    00 DD 0094C            PUSHL   #0
                 50 00000015G  EF   DO 0094E               MOVL    TEST+21,R0
                                    60 9F 00955            PUSHAB  (R0)
                 000000FF   8F      DD 00957               PUSHL   #255
                 00000000G  EF      9F 0095D               PUSHAB  JOURNAL_FILE
                                    05 FB 00963            CALLS   #5,PAS$WRITE_STRING
                 00000000G  EF      9F 0096A               PUSHAB  JOURNAL_FILE
                                    01 FB 00970            CALLS   #1,PAS$WRITELN2
                                    00V 11 00977           BRB     82$
                 00000000G  EF      9F 00979 80$:          PUSHAB  JOURNAL_FILE                        ; 6376
                                    01 FB 0097F            CALLS   #1,PAS$WRITELN2
                 07 00000019G  EF   91 00986 82$:          CMPB    TEST+25,#7                          ; 6378
                                    03 13 0098D            BEQL    .+3
                                0000V 31 0098F             BRW     90$
                 00000011G  EF      9F 00992               PUSHAB  TEST+17                             ; 6382
                 00000011G  EF      9F 00998               PUSHAB  TEST+17
                                    02 FB 0099E            CALLS   #2,STR$UPCASE
                 00000011G  EF      B5 009A5               TSTW    TEST+17                             ; 6384
                                    00V 12 009AB           BNEQ    85$
                 00000011G  EF      9F 009AD               PUSHAB  TEST+17                             ; 6388
                                    01 FB 009B3            CALLS   #1,STR$FREE1_DX
```

```
                                    00   DD 009BA          PUSHL   #0                                          ; 6389
                                    00   DD 009BC          PUSHL   #0
                                    00   DD 009BE          PUSHL   #0
                         00B38040   8F   DD 009C0          PUSHL   #11763776
    00000000G   EF                  04   FB 009C6          CALLS   #4,LIB$SIGNAL
                50                  01   D0 009CD 85$:     MOVL    #1,R0                                      ; 6397
                52 00000011G        EF   3C 009D0          MOVZWL  TEST+17,R2
                52                  50   D1 009D7          CMPL    R0,R2
                                    00V  14 009DA          BGTR    87$
          FC    AC                  50   D0 009DC 86$:     MOVL    R0,-4(R12)
                51        FC    AC  D0 009E0          MOVL    -4(R12),R1                                 ; 6399
                53        FC    AC  D0 009E4          MOVL    -4(R12),R3
                54 00000015G        EF   D0 009E8          MOVL    TEST+21,R4
          FEEC CC41         FF A443   90 009EF          MOVB    -1(R4)[R3],-276(R12)[R1]
E1              50                  52   F3 009F7          AOBLEQ  R2,R0,86$
                         00000000G  EF   9F 009FB 87$:     PUSHAB  QUAD_TIME                                  ; 6401
          F0    AD 010E00FF         8F   D0 00A01          MOVL    #17694975,-16(FP)
          F4    AD      FEED        CC   9E 00A09          MOVAB   -275(R12),-12(FP)
                           F0    AD  9F 00A0F          PUSHAB  -16(FP)
    00000000G   EF                  02   FB 00A12          CALLS   #2,SYS$BINTIM
                00V                 50   E8 00A19          BLBS    R0,90$
                         00000011G  EF   9F 00A1C          PUSHAB  TEST+17                                    ; 6407
    00000000G   EF                  01   FB 00A22          CALLS   #1,STR$FREE1_DX
                                    00   DD 00A29          PUSHL   #0                                         ; 6408
                                    00   DD 00A2B          PUSHL   #0
                                    00   DD 00A2D          PUSHL   #0
                         00B38030   8F   DD 00A2F          PUSHL   #11763760
    00000000G   EF                  04   FB 00A35          CALLS   #4,LIB$SIGNAL
          81    8F 0000001EG        EF   91 00A3C 90$:     CMPB    TEST+30,#-127                              ; 6414
                                    00V  12 00A44          BNEQ    92$
                20 00000011G        EF   B1 00A46          CMPW    TEST+17,#32
                                    00V  1A 00A4D          BGTRU   93$
          007E  8F 00000011G        EF   B1 00A4F 92$:     CMPW    TEST+17,#126
                                    00V  1B 00A58          BLEQU   95$
                         00000011G  EF   9F 00A5A 93$:     PUSHAB  TEST+17                                    ; 6426
    00000000G   EF                  01   FB 00A60          CALLS   #1,STR$FREE1_DX
                                    00   DD 00A67          PUSHL   #0                                         ; 6427
                                    00   DD 00A69          PUSHL   #0
                                    00   DD 00A6B          PUSHL   #0
                         00B38030   8F   DD 00A6D          PUSHL   #11763760
    00000000G   EF                  04   FB 00A73          CALLS   #4,LIB$SIGNAL
                50 0000001EG        EF   9A 00A7A 95$:     MOVZBL  TEST+30,R0                                 ; 6433
                50                  04   C4 00A81          MULL2   #4,R0
                50                  03   C0 00A84          ADDL2   #3,R0
    03 00000000G   EF               50   E0 00A87          BBS     R0,SEC_TYPE..+3
                              0000V 31   00A8F          BRW     112$
                         FFFF5AC4   EF   9F 00A92          PUSHAB  C.AVL                                      ; 6437
                                    07   DD 00A98          PUSHL   #7
                         00000000G  EF   9F 00A9A          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                  03   FB 00AA0          CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF   9F 00AA7          PUSHAB  ANSI_REVERSE
                                    04   DD 00AAD          PUSHL   #4
                         00000000G  EF   9F 00AAF          PUSHAB  PAS$FV_OUTPUT
    00000000G   EF                  03   FB 00AB5          CALLS   #3,PAS$WRITE_STRING
                         FFFF5AA2   EF   9F 00ABC          PUSHAB  C.AVM
                                    03   DD 00AC2          PUSHL   #3
                         00000000G  EF   9F 00AC4          PUSHAB  PAS$FV_OUTPUT
```

```
                00000000G  EF                    03  FB 00ACA          CALLS   #3,PAS$WRITE_STRING
                                 00000000G  EF    9F  00AD1          PUSHAB  ANSI_RESET
                                            04    DD  00AD7          PUSHL   #4
                                 00000000G  EF    9F  00AD9          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                    03  FB 00ADF          CALLS   #3,PAS$WRITE_STRING
                                 FFFF5A7C   EF    9F  00AE6          PUSHAB  C.AVN
                                            03    DD  00AEC          PUSHL   #3
                                 00000000G  EF    9F  00AEE          PUSHAB  PAS$FV_OUTPUT
                00000000G  EF                    03  FB 00AF4          CALLS   #3,PAS$WRITE_STRING
                                 00000000   8F    DF  00AFB          PUSHAL  #0                                      ; 6438
                                       00   8F    9F  00B01          PUSHAB  #0
                      F4   AD  00000000G  EF    9E  00B04          MOVAB   EDF$AB_YES_NO_TABLE_STA,-12(FP)
                                       F4   AD    9F  00B0C          PUSHAB  -12(FP)
                      F0   AD  00000000G  EF    9E  00B0F          MOVAB   EDF$AB_YES_NO_TABLE_KEY,-16(FP)
                                       F0   AD    9F  00B17          PUSHAB  -16(FP)
                00000000G  EF                    04  FB 00B1A          CALLS   #4,PARSE_INPUT
                                            50    94  00B21          CLRB    R0                                      ; 6445
                           01   00000000G  EF    D1  00B23          CMPL    INPUT_VALUE,#1
                                            00V   12  00B2A          BNEQ    98$
                                            50    96  00B2C          INCB    R0
                0000002BG  EF                    50  90 00B2E 98$:  MOVB    R0,TEST+43
                           0B   00000019G  EF    91  00B35          CMPB    TEST+25,#11                             ; 6447
                                            00V   12  00B3C          BNEQ    103$
                                 0000001AG  EF    D5  00B3E          TSTL    TEST+26
                                            00V   12  00B44          BNEQ    103$
                      77   8F   0000001EG  EF    91  00B46          CMPB    TEST+30,#119
                                            00V   12  00B4E          BNEQ    103$
           00V0000002BG  EF                    00  E1  00B50          BBC     #0,TEST+43,103$
                                            00    DD  00B58          PUSHL   #0                                      ; 6457
                                            00    DD  00B5A          PUSHL   #0
                                            00    DD  00B5C          PUSHL   #0
                                 00B38038   8F    DD  00B5E          PUSHL   #11763768
                00000000G  EF                    04  FB 00B64          CALLS   #4,LIB$SIGNAL
                           0C   00000019G  EF    91  00B6B 103$:    CMPB    TEST+25,#12                             ; 6459
                                            00V   12  00B72          BNEQ    112$
                      88   8F   0000001EG  EF    91  00B74          CMPB    TEST+30,#-120
                                            00V   12  00B7C          BNEQ    112$
           00V0000002BG  EF                    00  E1  00B7E          BBC     #0,TEST+43,112$
                                 00000000   8F    DF  00B86          PUSHAL  #0                                      ; 6469
                                       62   8F    9F  00B8C          PUSHAB  #98
                                 00000000   8F    DF  00B8F          PUSHAL  #0
                                       08   8F    9F  00B95          PUSHAB  #8
                                       01   8F    9F  00B98          PUSHAB  #1
                00000000G  EF                    05  FB 00B9B          CALLS   #5,FIND_OBJECT
                           00V                   50  E9 00BA2          BLBC    R0,112$
                           50   00000000G  EF    D0  00BA5          MOVL    DEF_CURRENT,R0                          ; 6473
                           1D         23   A0    D1  00BAC          CMPL    35(R0),#29
                                            00V   13  00BB0          BEQL    112$
                                            00    DD  00BB2          PUSHL   #0                                      ; 6475
                                            00    DD  00BB4          PUSHL   #0
                                            00    DD  00BB6          PUSHL   #0
                                 00B38038   8F    DD  00BB8          PUSHL   #11763768
                00000000G  EF                    04  FB 00BBE          CALLS   #4,LIB$SIGNAL
                      63   8F   0000001EG  EF    91  00BC5 112$:    CMPB    TEST+30,#99                             ; 6483
                                            03    13  00BCD          BEQL    .+3
                                          0000V   31  00BCF          BRW     114$
                                 00000000G  EF    D4  00BD2          CLRL    EDF$GL_OWNER_UIC                        ; 6487
```

```
                                Generated Code

                                        FFFF598E   EF   9F 00BD8          PUSHAB  C.AVO                                   ; 6489
                                                   08   DD 00BDE          PUSHL   #8
                                        00000000G  EF   9F 00BE0          PUSHAB  PAS$FV_OUTPUT
                                00000000G  EF           03   FB 00BE6     CALLS   #3,PAS$WRITE_STRING
                                        00000000G  EF   9F 00BED          PUSHAB  ANSI_REVERSE
                                                   04   DD 00BF3          PUSHL   #4
                                        00000000G  EF   9F 00BF5          PUSHAB  PAS$FV_OUTPUT
                                00000000G  EF           03   FB 00BFB     CALLS   #3,PAS$WRITE_STRING
                                        FFFF596C   EF   9F 00C02          PUSHAB  C.AVP
                                                   03   DD 00C08          PUSHL   #3
                                        00000000G  EF   9F 00C0A          PUSHAB  PAS$FV_OUTPUT
                                00000000G  EF           03   FB 00C10     CALLS   #3,PAS$WRITE_STRING
                                        00000000G  EF   9F 00C17          PUSHAB  ANSI_RESET
                                                   04   DD 00C1D          PUSHL   #4
                                        00000000G  EF   9F 00C1F          PUSHAB  PAS$FV_OUTPUT
                                00000000G  EF           03   FB 00C25     CALLS   #3,PAS$WRITE_STRING
                                        FFFF5946   EF   9F 00C2C          PUSHAB  C.AVQ
                                                   03   DD 00C32          PUSHL   #3
                                        00000000G  EF   9F 00C34          PUSHAB  PAS$FV_OUTPUT
                                00000000G  EF           03   FB 00C3A     CALLS   #3,PAS$WRITE_STRING
                                        00000000   8F   DF 00C41          PUSHAL  #0                                      ; 6490
                                              00   8F   9F 00C47          PUSHAB  #0
                              F4   AD 00000000G  EF   9E 00C4A          MOVAB   EDF$AB_UIC_TABLE_STA,-12(FP)
                                        F4   AD   9F 00C52               PUSHAB  -12(FP)
                              F0   AD 00000000G  EF   9E 00C55          MOVAB   EDF$AB_UIC_TABLE_KEY,-16(FP)
                                        F0   AD   9F 00C5D               PUSHAB  -16(FP)
                00000000G  EF                   04   FB 00C60          CALLS   #4,PARSE_INPUT
                0000002CG  EF 00000000G  EF     D0 00C67          MOVL    EDF$GL_OWNER_UIC,TEST+44                        ; 6497
                      65   8F 0000001EG  EF     91 00C72 114$:    CMPB    TEST+30,#101                                    ; 6501
                                                03   13 00C7A          BEQL    .+3
                                              0000V  31 00C7C          BRW     117$
                                                50   D4 00C7F          CLRL    R0                                      ; 6505
                                    FC   AC           50   D0 00C81 116$:    MOVL    R0,-4(R12)
00000000G  EF                 01    FC   AC           00   F0 00C85          INSV    #0,-4(R12),#1,EDF$GL_PROT_MASK          ; 6507
                              EE              50      1F   F3 00C8F          AOBLEQ  #31,R0,116$
                                                FFFF58E3   EF   9F 00C93     PUSHAB  C.AVR                                   ; 6509
                                                           09   DD 00C99     PUSHL   #9
                                                00000000G  EF   9F 00C9B     PUSHAB  PAS$FV_OUTPUT
                                        00000000G  EF           03   FB 00CA1 CALLS   #3,PAS$WRITE_STRING
                                                00000000G  EF   9F 00CA8     PUSHAB  ANSI_REVERSE
                                                           04   DD 00CAE     PUSHL   #4
                                                00000000G  EF   9F 00CB0     PUSHAB  PAS$FV_OUTPUT
                                        00000000G  EF           03   FB 00CB6 CALLS   #3,PAS$WRITE_STRING
                                                FFFF58C5   EF   9F 00CBD     PUSHAB  C.AVS
                                                           03   DD 00CC3     PUSHL   #3
                                                00000000G  EF   9F 00CC5     PUSHAB  PAS$FV_OUTPUT
                                        00000000G  EF           03   FB 00CCB CALLS   #3,PAS$WRITE_STRING
                                                00000000G  EF   9F 00CD2     PUSHAB  ANSI_RESET
                                                           04   DD 00CD8     PUSHL   #4
                                                00000000G  EF   9F 00CDA     PUSHAB  PAS$FV_OUTPUT
                                        00000000G  EF           03   FB 00CE0 CALLS   #3,PAS$WRITE_STRING
                                                00000000G  EF   9F 00CE7     PUSHAB  CRLF_SHIFT
                                                           06   DD 00CED     PUSHL   #6
                                                00000000G  EF   9F 00CEF     PUSHAB  PAS$FV_OUTPUT
                                        00000000G  EF           03   FB 00CF5 CALLS   #3,PAS$WRITE_STRING
                                                FFFF588A   EF   9F 00CFC     PUSHAB  C.AVT
                                                           02   DD 00D02     PUSHL   #2
```

```
                00000000G   EF   9F 00D04          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 00D0A          CALLS   #3,PAS$WRITE_STRING
                00000000    8F   DF 00D11          PUSHAL  #0                                      ; 6510
                      00    8F   9F 00D17          PUSHAB  #0
      F4   AD 00000000G     EF   9E 00D1A          MOVAB   EDF$AB_PROT_TABLE_STA,-12(FP)
                      F4    AD   9F 00D22          PUSHAB  -12(FP)
      F0   AD 00000000G     EF   9E 00D25          MOVAB   EDF$AB_PROT_TABLE_KEY,-16(FP)
                      F0    AD   9F 00D2D          PUSHAB  -16(FP)
00000000G  EF               04   FB 00D30          CALLS   #4,PARSE_INPUT
00000030G  EF 00000000G     EF   D0 00D37          MOVL    EDF$GL_PROT_MASK,TEST+48                ; 6517
      21 0000001EG          EF   91 00D42 117$:    CMPB    TEST+30,#33                             ; 6521
                            03   13 00D49          BEQL    .+3
                          0000V  31 00D4B          BRW     137$
                FFFF583C    EF   9F 00D4E          PUSHAB  C.AVU                                   ; 6525
                            08   DD 00D54          PUSHL   #8
                00000000G   EF   9F 00D56          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 00D5C          CALLS   #3,PAS$WRITE_STRING
                00000000G   EF   9F 00D63          PUSHAB  ANSI_REVERSE
                            04   DD 00D69          PUSHL   #4
                00000000G   EF   9F 00D6B          PUSHAB  PAS$FV_OUTPUT
0000000UG  EF               03   FB 00D71          CALLS   #3,PAS$WRITE_STRING
                FFFF581A    EF   9F 00D78          PUSHAB  C.AVV
                            03   DD 00D7E          PUSHL   #3
                00000000G   EF   9F 00D80          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 00D86          CALLS   #3,PAS$WRITE_STRING
                00000000G   EF   9F 00D8D          PUSHAB  ANSI_RESET
                            04   DD 00D93          PUSHL   #4
                00000000G   EF   9F 00D95          PUSHAB  PAS$FV_OUTPUT
00000000G  EF               03   FB 00D9B          CALLS   #3,PAS$WRITE_STRING
                FFFF57F4    EF   9F 00DA2          PUSHAB  C.AVW
                            03   DD 00DA8          PUSHL   #3
                00000000G   EF   9F 00DAA          PUSHAB  PAS$FV_OUTPUT
   00000000G  EF               03   FB 00DB0       CALLS   #3,PAS$WRITE_STRING
                   00000000 8F   DF 00DB7          PUSHAL  #0                                      ; 6526
                         00 8F   9F 00DBD          PUSHAB  #0
         F4   AD 00000000G  EF   9E 00DC0          MOVAB   EDF$AB_POSIT_TABLE_STA,-12(FP)
                         F4 AD   9F 00DC8          PUSHAB  -12(FP)
         F0   AD 00000000G  EF   9E 00DCB          MOVAB   EDF$AB_POSIT_TABLE_KEY,-16(FP)
                         F0 AD   9F 00DD3          PUSHAB  -16(FP)
   00000000G  EF               04   FB 00DD6       CALLS   #4,PARSE_INPUT
   00000023G  EF 00000000G     EF   D0 00DDD       MOVL    INPUT_VALUE,TEST+35                     ; 6533
   00000100   8F 00000023G     EF   D1 00DE8       CMPL    TEST+35,#256                            ; 6535
                               00V  1E 00DF3       BGEQU   119$
03 FFFF579F   EF 00000023G     EF   E1 00DF5       BBC     TEST+35,C.AVX,..+3
                             0000V  31 00E01       BRW     136$
                   00000000G   EF   9F 00E04 119$: PUSHAB  CRLF_SHIFT                              ; 6539
                               06   DD 00E0A       PUSHL   #6
                   00000000G   EF   9F 00E0C       PUSHAB  PAS$FV_OUTPUT
   00000000G  EF               03   FB 00E12       CALLS   #3,PAS$WRITE_STRING
                   FFFF57A1    EF   9F 00E19       PUSHAB  C.AVY
                               17   DD 00E1F       PUSHL   #23
                   00000000G   EF   9F 00E21       PUSHAB  PAS$FV_OUTPUT
   00000000G  EF               03   FB 00E27       CALLS   #3,PAS$WRITE_STRING
06                 01 00000023G  EF   CF 00E2E    CASEL   TEST+35,#1,#6                           ; 6541
                                0000V     00E36    .DISPL  120$
                                0000V     00E38    .DISPL  120$
                                0000V     00E3A    .DISPL  121$
```

```
                                0000V     00E3C          .DISPL  122$
                                0000V     00E3E          .DISPL  120$
                                000E      00E40          .DISPL  14
                                0000V     00E42          .DISPL  120$
                                0000V  31 00E44          BRW     134$
                         FFFF578B   EF    9F 00E47 120$:  PUSHAB  C.AVZ                                    ; 6547
                                    08    DD 00E4D        PUSHL   #8
                         00000000G  EF    9F 00E4F        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00E55        CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF    9F 00E5C        PUSHAB  ANSI_REVERSE
                                    04    DD 00E62        PUSHL   #4
                         00000000G  EF    9F 00E64        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00E6A        CALLS   #3,PAS$WRITE_STRING
                         FFFF5769   EF    9F 00E71        PUSHAB  C.AWA
                                    03    DD 00E77        PUSHL   #3
                         00000000G  EF    9F 00E79        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00E7F        CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF    9F 00E86        PUSHAB  ANSI_RESET
                                    04    DD 00E8C        PUSHL   #4
                         00000000G  EF    9F 00E8E        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00E94        CALLS   #3,PAS$WRITE_STRING
                         FFFF5743   EF    9F 00E9B        PUSHAB  C.AWB
                                    03    DD 00EA1        PUSHL   #3
                         00000000G  EF    9F 00EA3        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00EA9        CALLS   #3,PAS$WRITE_STRING
                         00000000   8F    DF 00EB0        PUSHAL  #0                                       ; 6548
                                 00    8F    9F 00EB6        PUSHAB  #0
                         00000027G  EF    9F 00EB9        PUSHAB  TEST+39
00000000G  EF                       03    FB 00EBF        CALLS   #3,NUMBER_INPUT
                                0000V  31 00EC6          BRW     136$
                         00000000G  EF    D4 00EC9 121$:  CLRL    EDF$GL_FID1                              ; 6556
                         00000000G  EF    D4 00ECF        CLRL    EDF$GL_FID2                              ; 6557
                         00000000G  EF    D4 00ED5        CLRL    EDF$GL_FID3                              ; 6558
                         FFFF5707   EF    9F 00EDB        PUSHAB  C.AWC                                    ; 6560
                                    08    DD 00EE1        PUSHL   #8
                         00000000G  EF    9F 00EE3        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00EE9        CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF    9F 00EF0        PUSHAB  ANSI_REVERSE
                                    04    DD 00EF6        PUSHL   #4
                         00000000G  EF    9F 00EF8        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00EFE        CALLS   #3,PAS$WRITE_STRING
                         FFFF56E5   EF    9F 00F05        PUSHAB  C.AWD
                                    03    DD 00F0B        PUSHL   #3
                         00000000G  EF    9F 00F0D        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00F13        CALLS   #3,PAS$WRITE_STRING
                         00000000G  EF    9F 00F1A        PUSHAB  ANSI_RESET
                                    04    DD 00F20        PUSHL   #4
                         00000000G  EF    9F 00F22        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00F28        CALLS   #3,PAS$WRITE_STRING
                         FFFF56BF   EF    9F 00F2F        PUSHAB  C.AWE
                                    03    DD 00F35        PUSHL   #3
                         00000000G  EF    9F 00F37        PUSHAB  PAS$FV_OUTPUT
00000000G  EF                       03    FB 00F3D        CALLS   #3,PAS$WRITE_STRING
                         00000000   8F    DF 00F44        PUSHAL  #0                                       ; 6561
                                 00    8F    9F 00F4A        PUSHAB  #0
                F4    AD 00000000G  EF    9E 00F4D        MOVAB   EDF$AB_FID_TABLE_STA,-12(FP)
                                 F4    AD    9F 00F55        PUSHAB  -12(FP)
```

```
                 FO   AD 00000000G   EF   9E 00F58          MOVAB   EDF$AB_FID_TABLE_KEY,-16(FP)
                         FO   AD     9F   00F60          PUSHAB  -16(FP)
          00000000G   EF             04   FB 00F63          CALLS   #4,PARSE_INPUT
          00000034G   EF 00000000G   EF   D0 00F6A          MOVL    EDF$GL_FID1,TEST+52                    ; 6568
          00000038G   EF 00000000G   EF   D0 00F75          MOVL    EDF$GL_FID2,TEST+56                    ; 6569
          0000003CG   EF 00000000G   EF   D0 00F80          MOVL    EDF$GL_FID3,TEST+60                    ; 6570
                               0000V 31   00F8B          BRW     136$
                             FFFF5664 EF   9F 00F8E 122$:   PUSHAB  C.AWF                                  ; 6578
                                      12   DD 00F94          PUSHL   #18
                             00000000G EF   9F 00F96          PUSHAB  PAS$FV_OUTPUT
          00000000G   EF             03   FB 00F9C          CALLS   #3,PAS$WRITE_STRING
                             00000000G EF   9F 00FA3          PUSHAB  CRLF_SHIFT
                                      06   DD 00FA9          PUSHL   #6
                             00000000G EF   9F 00FAB          PUSHAB  PAS$FV_OUTPUT
          00000000G   EF             03   FB 00FB1          CALLS   #3,PAS$WRITE_STRING
                             FFFF564C EF   9F 00FB8          PUSHAB  C.AWG
                                      02   DD 00FBE          PUSHL   #2
                             00000000G EF   9F 00FC0          PUSHAB  PAS$FV_OUTPUT
          00000000G   EF             03   FB 00FC6          CALLS   #3,PAS$WRITE_STRING
       00V00000000G   EF             30   E0 00FCD          BBS     #48,PAS$FV_INPUT,123$                  ; 6580
                             00000000G EF   9F 00FD5          PUSHAB  PAS$FV_INPUT
          00000000G   EF             01   FB 00FDB          CALLS   #1,PAS$LOOK_AHEAD
       00V00000000G   EF             31   E0 00FE2 123$:    BBS     #49,PAS$FV_INPUT,125$
                             00000000G EF   9F 00FEA          PUSHAB  PAS$FV_INPUT                           ; 6584
          00000000G   EF             01   FB 00FF0          CALLS   #1,PAS$RESET2
                                      00   DD 00FF7          PUSHL   #0                                     ; 6585
                                      00   DD 00FF9          PUSHL   #0
                                      00   DD 00FFB          PUSHL   #0
                                00B3804B   8F   DD 00FFD          PUSHL   #11763787
          00000000G   EF             04   FB 01003          CALLS   #4,LIB$SIGNAL
                                000000FF   8F   DD 0100A 125$:    PUSHL   #255                               ; 6589
                                00000000G  EF   9F 01010          PUSHAB  PAS$FV_INPUT
                                    FEED   CC   9F 01016          PUSHAB  -275(R12)
          00000000G   EF             03   FB 0101A          CALLS   #3,PAS$READ_STRING
                             00000000G EF   9F 01021          PUSHAB  PAS$FV_INPUT
          00000000G   EF             01   FB 01027          CALLS   #1,PAS$READIN2
                             00000000G EF   9F 0102E          PUSHAB  CRLF                                   ; 6590
                                      02   DD 01034          PUSHL   #2
                             00000000G EF   9F 01036          PUSHAB  PAS$FV_OUTPUT
          00000000G   EF             03   FB 0103C          CALLS   #3,PAS$WRITE_STRING
                             00000000G EF   9F 01043          PUSHAB  PAS$FV_OUTPUT
          00000000G   EF             01   FB 01049          CALLS   #1,PAS$WRITELN2
          00000011G   EF 00000000G   EF   7D 01050          MOVQ    NULL_STRING,TEST+17                    ; 6592
                 F0   AD 010E00FF    8F   D0 0105B          MOVL    #17694975,-16(FP)                      ; 6593
                 F4   AD     FEED    CC   9E 01063          MOVAB   -275(R12),-12(FP)
                             FO      AD   9F 01069          PUSHAB  -16(FP)
                             000000116 EF   9F 0106C          PUSHAB  TEST+17
          00000000G   EF             02   FB 01072          CALLS   #2,STR$TRIM
                             00000000G EF   9F 01079          PUSHAB  INPUT_DESC                             ; 6594
                             000000116 EF   9F 0107F          PUSHAB  TEST+17
          00000000G   EF             02   FB 01085          CALLS   #2,LIB$SCOPY_DXDX
          00000014G   EF 00000004G   EF   D0 0108C          MOVL    INPUT_DESC+4,PARAM_BLOCK+20            ; 6595
          00000010G   EF 00000000G   EF   3C 01097          MOVZWL  INPUT_DESC,PARAM_BLOCK+16              ; 6596
       00V00000000G   EF             00   E1 010A2          BBC     #0,JOURNAL_ENABLED,131$                ; 6602
                             00000011G EF   B5 010AA          TSTW    TEST+17                                ; 6604
                                      00V  1B 010B0          BLEQU   129$
                                7E 00000011G EF   3C 010B2          MOVZWL  TEST+17,-(SP)                  ; 6606
```

```
                                  00    DD 010B9          PUSHL   #0
                            50 00000015G  EF    D0 010BB          MOVL    TEST+21,R0
                                  60    9F 010C2          PUSHAB  (R0)
                               000000FF   8F    DD 010C4          PUSHL   #255
                               00000000G  EF    9F 010CA          PUSHAB  JOURNAL_FILE
                00000000G   EF            05    FB 010D0          CALLS   #5,PAS$WRITE_STRING
                               00000000G  EF    9F 010D7          PUSHAB  JOURNAL_FILE
                00000000G   EF            01    FB 010DD          CALLS   #1,PAS$WRITELN2
                                  00V 11 010E4          BRB     130$
                               00000000G  EF    9F 010E6 129$:    PUSHAB  JOURNAL_FILE                    ; 6614
                00000000G   EF            01    FB 010EC          CALLS   #1,PAS$WRITELN2
                                             010F3 130$:
                    006D    8F 00000011G  EF    B1 010F3 131$:    CMPW    TEST+17,#109                    ; 6616
                                  00V 1B 010FC          BLEQU   136$
                                  00    DD 010FE          PUSHL   #0                              ; 6618
                                  00    DD 01100          PUSHL   #0
                                  00    DD 01102          PUSHL   #0
                               00B38030   8F    DD 01104          PUSHL   #11763760
                00000000G   EF            04    FB 0110A          CALLS   #4,LIB$SIGNAL
                                  00V 11 01111          BRB     136$
                                             01113 134$:
                                             01113 136$:
                      83    8F 0000001EG  EF    91 01113 137$:    CMPB    TEST+30,#-125                   ; 6632
                                  00V 13 0111B          BEQL    139$
                      59    8F 0000001EG  EF    91 0111D          CMPB    TEST+30,#89
                                  03    13 01125          BEQL    .+3
                                  0000V 31 01127          BRW     166$
                               FFFF54DC   EF    9F 0112A 139$:    PUSHAB  C.AWH                           ; 6640
                                  0B    DD 01130          PUSHL   #11
                               00000000G  EF    9F 01132          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF            03    FB 01138          CALLS   #3,PAS$WRITE_STRING
                               00000000G  EF    9F 0113F          PUSHAB  ANSI_REVERSE
                                  04    DD 01145          PUSHL   #4
                               00000000G  EF    9F 01147          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF            03    FB 0114D          CALLS   #3,PAS$WRITE_STRING
                               FFFF54BE   EF    9F 01154          PUSHAB  C.AWI
                                  03    DD 0115A          PUSHL   #3
                               00000000G  EF    9F 0115C          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF            03    FB 01162          CALLS   #3,PAS$WRITE_STRING
                               00000000G  EF    9F 01169          PUSHAB  ANSI_RESET
                                  04    DD 0116F          PUSHL   #4
                               00000000G  EF    9F 01171          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF            03    FB 01177          CALLS   #3,PAS$WRITE_STRING
                               FFFF5498   EF    9F 0117E          PUSHAB  C.AWJ
                                  03    DD 01184          PUSHL   #3
                               00000000G  EF    9F 01186          PUSHAB  PAS$FV_OUTPUT
                00000000G   EF            03    FB 0118C          CALLS   #3,PAS$WRITE_STRING
             00V00000000G   EF            30    E0 01193          BBS     #48,PAS$FV_INPUT,140$           ; 6643
                               00000000G  EF    9F 0119B          PUSHAB  PAS$FV_INPUT
                00000000G   EF            01    FB 011A1          CALLS   #1,PAS$LOOK_AHEAD
             00V00000000G   EF            31    E0 011A8 140$:    BBS     #49,PAS$FV_INPUT,142$
                               00000000G  EF    9F 011B0          PUSHAB  PAS$FV_INPUT                    ; 6647
                00000000G   EF            01    FB 011B6          CALLS   #1,PAS$RESET2
                                  00    DD 011BD          PUSHL   #0                              ; 6648
                                  00    DD 011BF          PUSHL   #0
                                  00    DD 011C1          PUSHL   #0
                               00B3804B   8F    DD 011C3          PUSHL   #11763787
```

```
                    00000000G  EF                  04   FB 011C9                CALLS   #4,LIB$SIGNAL
                                          000000FF  8F   DD 011D0 142$:          PUSHL   #255                                ; 6652
                                          00000000G EF   9F 011D6                PUSHAB  PAS$FV_INPUT
                                              FEED  CC   9F 011DC                PUSHAB  -275(R12)
                    00000000G  EF                  03   FB 011E0                CALLS   #3,PAS$READ_STRING
                                          00000000G EF   9F 011E7                PUSHAB  PAS$FV_INPUT
                    00000000G  EF                  01   FB 011ED                CALLS   #1,PAS$READLN2
                                          00000000G EF   9F 011F4                PUSHAB  CRLF                                ; 6653
                                                    02   DD 011FA                PUSHL   #2
                                          00000000G EF   9F 011FC                PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                  03   FB 01202                CALLS   #3,PAS$WRITE_STRING
                                          00000000G EF   9F 01209                PUSHAB  PAS$FV_OUTPUT
                    00000000G  EF                  01   FB 0120F                CALLS   #1,PAS$WRITELN2
                          F0   AC 00000000G        EF   7D 01216                MOVQ    NULL_STRING,-16(R12)                ; 6655
                          F0   AD 010E00FF         8F   D0 0121E                MOVL    #17694975,-16(FP)                   ; 6656
                          F4   AD     FEED         CC   9E 01226                MOVAB   -275(R12),-12(FP)
                                                F0   AD   9F 0122C                PUSHAB  -16(FP)
                                                F0   AC   9F 0122F                PUSHAB  -16(R12)
                    00000000G  EF                  02   FB 01232                CALLS   #2,STR$TRIM
                                          00000000G EF   9F 01239                PUSHAB  INPUT_DESC                          ; 6657
                                                F0   AC   9F 0123F                PUSHAB  -16(R12)
                    00000000G  EF                  02   FB 01242                CALLS   #2,LIB$SCOPY_DXDX
                    00000014G  EF 00000004G        EF   D0 01249                MOVL    INPUT_DESC+4,PARAM_BLOCK+20         ; 6658
                    00000010G  EF 00000000G        EF   3C 01254                MOVZWL  INPUT_DESC,PARAM_BLOCK+16           ; 6659
                00V000000000G  EF                  00   E1 0125F                BBC     #0,JOURNAL_ENABLED,148$             ; 6665
                                                F0   AC   B5 01267                TSTW    -16(R12)                            ; 6667
                                                   00V    15 0126A                BLEQ    146$
                                             7E    F0   AC   3C 0126C             MOVZWL  -16(R12),-(SP)                      ; 6669
                                                   00   DD 01270                PUSHL   #0
                                                F4   BC   9F 01272                PUSHAB  @-12(R12)
                                          000000FF  8F   DD 01275                PUSHL   #255
                                          00000000G EF   9F 0127B                PUSHAB  JOURNAL_FILE
                    00000000G  EF                  05   FB 01281                CALLS   #5,PAS$QRITE_STRING
                                          00000000G EF   9F 01288                PUSHAB  JOURNAL_FILE
                    00000000G  EF                  01   FB 0128E                CALLS   #1,PAS$QRITELN2
                                                   00V    11 01295                BRB     148$
                                          00000000G EF   9F 01297 146$:          PUSHAB  JOURNAL_FILE                        ; 6677
                    00000000G  EF                  01   FB 0129D                CALLS   #1,PAS$QRITELN2
                                                F0   AC   B5 012A4 148$:          TSTW    -16(R12)                            ; 6679
                                                   00V    12 012A7                BNEQ    150$
                                                F0   AC   9F 012A9                PUSHAB  -16(R12)                            ; 6683
                    00000000G  EF                  01   FB 012AC                CALLS   #1,STR$FREE1_DX
                                                   00   DD 012B3                PUSHL   #0                                  ; 6684
                                                   00   DD 012B5                PUSHL   #0
                                                   00   DD 012B7                PUSHL   #0
                                          00B38040  8F   DD 012B9                PUSHL   #11763776
                    00000000G  EF                  04   FB 012BF                CALLS   #4,LIB$SIGNAL
                                                F8   AC   9F 012C6 150$:          PUSHAB  -8(R12)                             ; 6688
                                                F0   AC   9F 012C9                PUSHAB  -16(R12)
                    00000000G  EF                  02   FB 012CC                CALLS   #2,OTS$CVT_TI_L
                    00000000G  EF                  50   D0 012D3                MOVL    R0,ISTATUS
                            00V000000000G          EF   E9 012DA                BLBC    ISTATUS,152$                        ; 6690
                    00000027G  EF           F8     AC   D0 012E1                MOVL    -8(R12),TEST+39                     ; 6692
                                                   00V    11 012E9                BRB     157$
                    00000000G  EF           F4     BC   91 012EB 152$:          CMPB    @-12(R12),APOSTROPHE                ; 6694
                                                   00V    12 012F3                BNEQ    154$
                                             50    F4   AC   D0 012F5             MOVL    -12(R12),R0
```

```
                                                Generated Code

                                00000000G  EF        02    AO   91 012F9          CMPB    2(R0),APOSTROPHE
                                                          00V  13 01301          BEQL    155$
                                                       F0    AC   9F 01303 154$:  PUSHAB  -16(R12)                                ; 6702
                                00000000G  EF              01   FB 01306          CALLS   #1,STR$FREE1_DX
                                                          00   DD 0130D          PUSHL   #0                                   ; 6703
                                                          00   DD 0130F          PUSHL   #0
                                                          00   DD 01311          PUSHL   #0
                                                00B38030  8F   DD 01313          PUSHL   #11763760
                                00000000G  EF             04   FB 01319          CALLS   #4,LIB$SIGNAL
                                                          00V  11 01320          BRB     157$
                                                    50    F4   AC   D0 01322 155$: MOVL    -12(R12),R0                         ; 6709
                                00000027G  EF       01    AO   9A 01326          MOVZBL  1(R0),TEST+39
                                        59  8F 0000001EG  EF   91 0132E 157$:     CMPB    TEST+30,#89                         ; 6711
                                                          00V  12 01336          BNEQ    165$
                                                    20 00000027G  EF   D1 01338  CMPL    TEST+39,#32                         ; 6718
                                                          00V  19 0133F          BLSS    163$
00000027G  EF       5A  8F                          07    00   ED 01341          CMPZV   #0,#7,#^X5A,TEST+39
                                                          00V  19 0134B          BLSS    163$
                                                    23 00000027G  EF   D1 0134D  CMPL    TEST+39,#35
                                                          00V  13 01354          BEQL    163$
                                                    24 00000027G  EF   D1 01356  CMPL    TEST+39,#36
                                                          00V  13 0135D          BEQL    163$
00000027G  EF       40  8F                          07    00   ED 0135F          CMPZV   #0,#7,#^X40,TEST+39
                                                          00V  12 01369          BNEQ    164$
                                                       F0    AC   9F 0136B 163$:  PUSHAB  -16(R12)                                ; 6732
                                00000000G  EF              01   FB 0136E          CALLS   #1,STR$FREE1_DX
                                                          00   DD 01375          PUSHL   #0                                   ; 6733
                                                          00   DD 01377          PUSHL   #0
                                                          00   DD 01379          PUSHL   #0
                                                00B38038  8F   DD 0137B          PUSHL   #11763768
                                00000000G  EF             04   FB 01381          CALLS   #4,LIB$SIGNAL
                                                                01388 164$:
                                                                01388 165$:
                                        00000000G  EF    94    01388 166$:      CLRB    TEMP_FULL_PROMPT                    ; 6744
                                                          04    0138E            RET                                         ; 6746

; Routine Size: 5007 bytes,    Routine Base: $CODE + 0D880

                                                                0ECOF            .END
```

COMMAND QUALIFIERS

```
  PASCAL/MACHINE/NODEBUG/NOCHECK/LIS=LIS$:EDFASK/OBJ=OBJ$:EDFASK MSRC$:EDFASK

  /CHECK=(NOBOUNDS,NOCASE_SELECTORS,NOOVERFLOW,NOPOINTERS,NOSUBRANGE)
  /DEBUG=(NOSYMBOLS,NOTRACEBACK)
  /ENVIRONMENT= $255$DUA28:[EDF.OBJ]EDFASK.PEN;1
  /LIST= $255$DUA28:[EDF.LIS]EDFASK.LIS;1
  /OBJECT= $255$DUA28:[EDF.OBJ]EDFASK.OBJ;1
  /NOCROSS_REFERENCE /ERROR_LIMIT=30 /NOG_FLOATING /MACHINE_CODE /NOOLD_VERSION /OPTIMIZE /NOSTANDARD /WARNINGS
```

COMPILER INTERNAL TIMING

Phase	Faults	CPU Time	Elapsed Time
Initialization	89	00:00.5	00:02.5
Source Analysis	1199	00:31.6	05:00.3
Source Listing	50	00:08.2	00:16.4
Tree Construction	1182	00:06.2	00:12.5
Flow Analysis	111	00:02.9	00:05.2
Profit Analysis	62	00:03.3	00:07.0
Context Analysis	1121	00:37.2	01:11.1
Name Packing	67	00:01.2	00:02.0
Code Selection	809	00:09.0	00:19.1
Final	834	00:46.3	02:02.3
TOTAL	5531	02:26.5	09:18.5

COMPILATION STATISTICS

```
  CPU Time:       02:26.5         (2774 Lines/Minute)
  Elapsed Time:   09:18.5
  Page Faults:    5531
  Compilation Complete
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

EDF

EDFVALUE
PAS

EDF
MAP

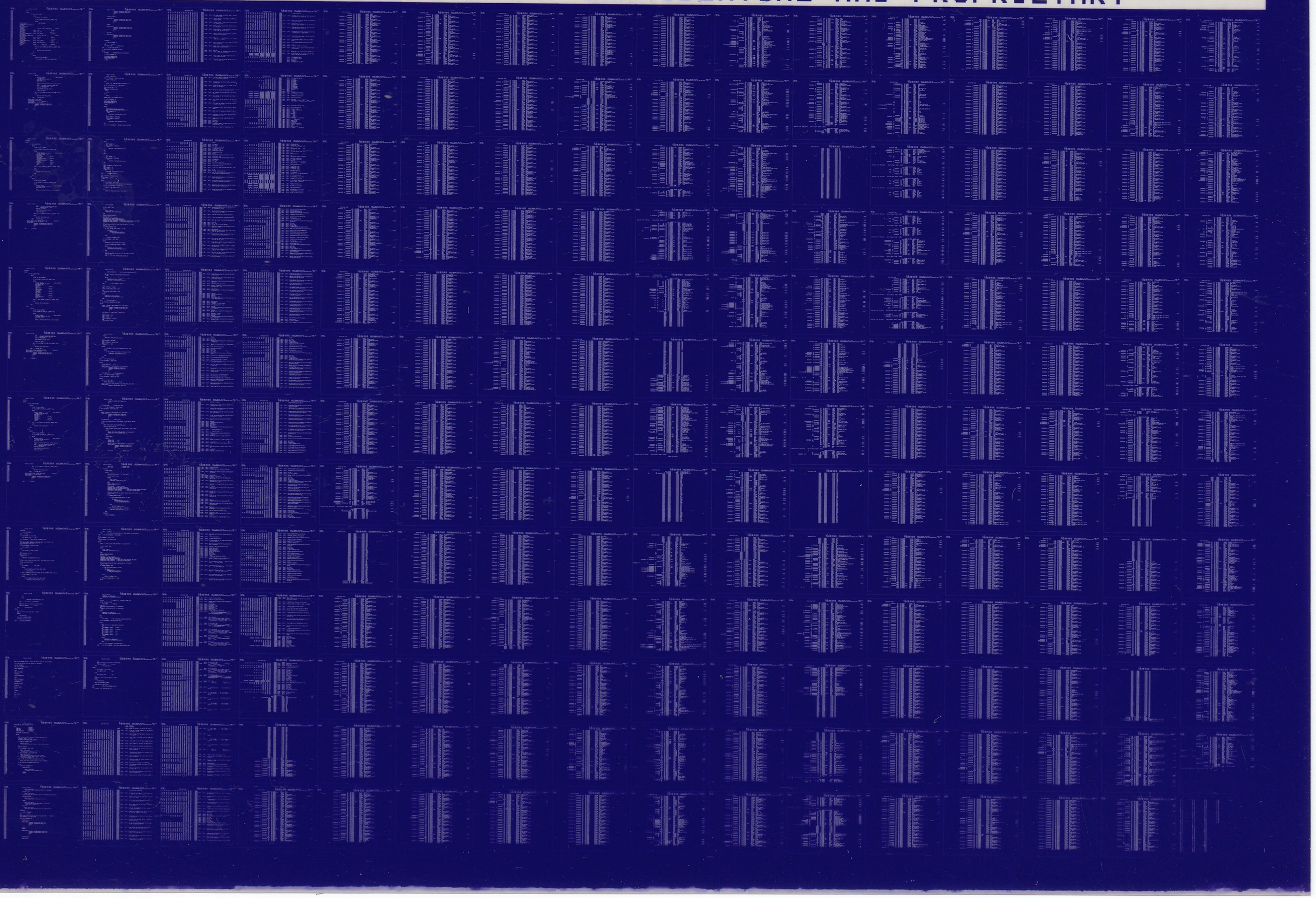
EDFSTRUCT
SDL

DUMPMSG
LIS

EDFASK
LIS

0125 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

EDFCHF
LIS

EDFFUNCS
LIS

EDFEXTERN
LIS

EDFDESIGN
LIS

EDFCONST
LIS